

1/5/1 (Item 1 from file: 351)
DIALOG(R)File 351:Derwent WPI
(c) 2006 The Thomson Corporation. All rts. reserv.

0008934067

WPI ACC NO: 1998-485662/ 19 98 42

XRPX Acc No: N1998-379207

Moving image search system - holds data of direction vector of motion, allowance colour display of colour area, number of pixels and centre of gravity coordinate of colour area

Patent Assignee: FUJITSU LTD (FUIT)

Inventor: ISHIKAWA H; KATOU K

Patent Family (2 patents, 1 countries)

Patent	Application			
Number	Kind	Date	Number	Kind
JP 10207897	A	19980807	JP 19977052	A
JP 3808153	B2	20060809	JP 19977052	A
				19970117
				199842 B
				200652 E

Priority Applications (no., kind, date): JP 19977052 A 19970117

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
JP 10207897	A	JA	61	43	
JP 3808153	B2	JA	43		Previously issued patent JP 10207897

Alerting Abstract JP A

The system has a controller (28) which extracts a body motion area from MPEG data. The motion area is divided to a colour area. A memory (25) holds the data of the direction vector of motion, allowance colour display of the colour area, number of pixels of each colour area, and centre of gravity coordinate of the colour area as an amount of the characteristics.

The search data are input by a keyboard (12). Based on the search data, amount of characteristics is searched. The image of the frame containing the body motion area is displayed in a display unit (13).

ADVANTAGE - Is uninfluenced by background other than body motion area.
Searches moving image effectively.

Title Terms/Index Terms/Additional Words: MOVE; IMAGE; SEARCH; SYSTEM; HOLD ; DATA; DIRECTION; VECTOR; MOTION; ALLOW; COLOUR; DISPLAY; AREA; NUMBER; PIXEL; CENTRE; GRAVITY; COORDINATE

Class Codes

International Classification (Main): G06F-017/30
(Additional/Secondary): G06T-013/00, G06T-007/20

International Classification (+ Attributes)

IPC + Level Value Position Status Version

G06F-0017/30	A	I	F	B	20060101
G06T-0007/20	A	I	L	B	20060101
H04N-0005/76	A	I	L	B	20060101

File Segment: EPI;

DWPI Class: T01

Manual Codes (EPI/S-X): T01-J05B3; T01-J10B2; T01-J10C5

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平10-207897

(43)公開日 平成10年(1998)8月7日

(51)Int.Cl.⁶

G 0 6 F 17/30
G 0 6 T 13/00
7/20

識別記号

F I

G 0 6 F 15/40 3 7 0 D
15/403 3 5 0 C
15/62 3 4 0 A
15/70 4 1 0

審査請求 未請求 請求項の数8 O.L (全 61 頁)

(21)出願番号

特願平9-7052

(22)出願日

平成9年(1997)1月17日

(71)出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中4丁目1番
1号

(72)発明者 加藤 光幾

神奈川県川崎市中原区上小田中4丁目1番
1号 富士通株式会社内

(72)発明者 石川 博

神奈川県川崎市中原区上小田中4丁目1番
1号 富士通株式会社内

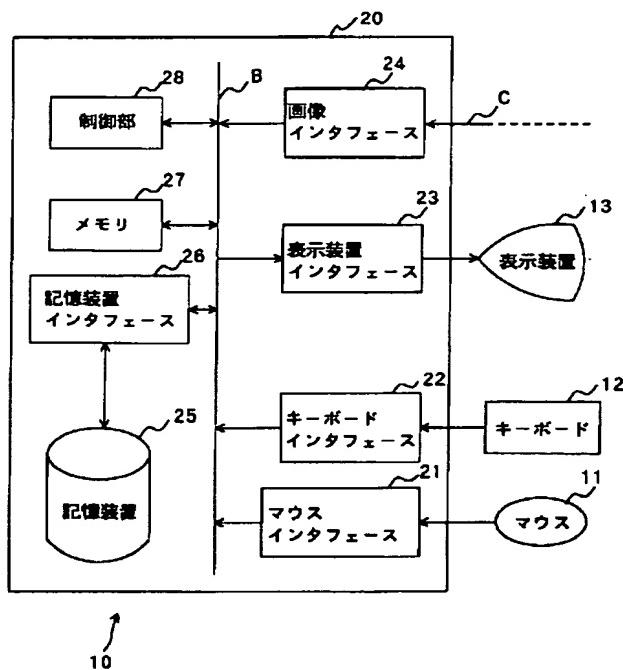
(74)代理人 弁理士 遠山 勉 (外1名)

(54)【発明の名称】 動画像検索システム

(57)【要約】

【課題】動画像データ中の動体部分の画像データに基づく動画像検索を行うことができる動画像検索システムを提供する。

【解決手段】制御部28によって、MPEGデータから動体領域が抽出される。制御部28によって、前記動体領域が色領域に分割される。記憶装置25によって、前記動体領域の移動方向ベクトルのデータ、前記色領域の代表色のデータ、前記色領域の面積のデータ、及び前記色領域の重心座標のデータが特微量データベースとして保持される。キーボード12によって、検索データが入力される。制御部28によって、前記検索データに基づいて前記特微量データベースが検索される。制御部28によって、前記動体領域を含むフレームの画像が表示装置13に表示される。



量データに基づいて前記特微量データベース手段を検索して前記フレーム特定データを得る特微量データベース検索手段と、

前記特微量データベース検索手段によって得られた前記フレーム特定データによって特定されるフレームを前記動画像データ保持手段から読み出して前記表示手段に表示させる制御手段とを備えたことを特徴とする動画像検索システム。

【請求項7】前記特微量データ抽出手段は、所定数の色彩を基準に前記動体領域を分割して1つ以上の色領域とし、前記色領域の大きさ及び色彩を前記特微量データとするものであり、前記特微量データ入力手段は、検索対象の大きさ及び色彩を入力するものであり、前記特微量データベース検索手段は、前記特微量データ入力手段によって入力された検索対象の大きさ及び色彩によって前記特微量データベース手段を検索するものであることを特徴とする請求項1乃至6の何れかに記載の動画像検索システム。

【請求項8】前記特微量データ抽出手段は、前記色領域の大きさ及び色彩に加えて、前記動体領域の移動方向ベクトルを前記特微量データとするものであり、前記特微量データ入力手段は、検索対象の大きさ、色彩及び移動方向ベクトルを入力するものであり、前記特微量データベース検索手段は、前記特微量データ入力手段によって入力された検索対象の大きさ、色彩及び移動方向ベクトルによって前記特微量データベース手段を検索するものであることを特徴とする請求項7記載の動画像検索システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、動画像データ中から所望のフレームを検索する動画像検索システムに関するものである。

【0002】

【従来の技術】従来、ユーザが動画像データ中から所望のフレームを検索するため、キーワード検索が行われていた。このような、従来のキーワード検索について説明する。まず、オペレータが、フレームやフレームを含むショットを特徴づけるようなキーワードを予めフレームやショットに付与しておく。そして、ユーザから検索のためのキーワードが入力されると、当該キーワードと一致するようなキーワードが付与されたフレームやショットを検索する。

【0003】このキーワード検索では、動画像のデータ量が膨大であるためキーワードをフレームやショットに付与する手間が膨大となっていた。また、オペレータがキーワードを付与するため、同一の動画像に付与されるキーワードが、オペレータによって異なってしまっていた。更に、動画像が含む全ての情報をキーワードによって表現することは、不可能であった。

【0004】キーワード検索には上述のような欠点があるため、静止画像検索を動画像検索に応用した動画像検索システムも、用いられている。このような、動画像検索システムについて説明する。まず、動画像を構成する各フレーム中の色彩のヒストグラム、直線・円等の形状、テクスチャ等を特微量として抽出し、各フレームに対応させておく。そして、ユーザから検索のための特微量が入力されると、当該特微量と一致するような特微量を対応させたフレームを検索する。

10 【0005】

【発明が解決しようとする課題】しかし、このような従来の動画像検索システムでは、以下のような欠点があった。即ち、従来の動画像検索システムにおいては、各フレームの画像全体から特微量を抽出していた。従って、背景がフレームの大部分を占めているような場合には、付与される特微量は殆ど背景に関するものとなる。そのため、ユーザが各フレーム中の物体を検索しようとして当該物体の特微量を入力しても、満足な検索結果を得られない場合があった。

20 【0006】

本発明の課題は、このような従来技術の問題点に鑑み、各フレーム中の物体のみの特微量に基づいて動画像検索を行うことができる動画像検索システムを提供することである。

【0007】

【課題を解決するための手段】本発明は前記課題を解決するために、以下の手段を採用した。本願の第1の発明は、上述した課題を解決するため、動画像データを保持する動画像データ保持手段と、前記動画像データに基づく動画像を表示する表示手段と、前記動画像データのフレーム中から動体部分の画像データである動体領域を抽出する動体領域抽出手段と、前記動体領域抽出手段によって抽出された動体領域を特徴づける特微量データを抽出する特微量データ抽出手段と、前記特微量データに対応させて前記動体領域を含むフレームを特定するフレーム特定データを保持する特微量データベース手段と、検索対象の動画像データの前記特微量データを入力するための特微量データ入力手段と、前記特微量データ入力手段によって入力された前記特微量データに基づいて前記特微量データベース手段を検索して前記フレーム特定データを得る特微量データベース検索手段と、前記特微量データベース検索手段によって得られた前記フレーム特定データによって特定されるフレームを前記動画像データ保持手段から読み出して前記表示手段に表示させる制御手段とを備えたことを特徴とする（請求項1に対応）。

30 【0008】

ここで、動画像データ保持手段、特微量データベース手段とはハードディスク等であり、表示手段とはCRT(Cathode Ray Tube)等であり、動体領域抽出手段、特微量データ抽出手段、特微量データベース検索手段、制御手段とはCPU(Cen

tral Processing Unit), DSP (Digital Signal Processor) 等であり、特微量データ入力手段とはマウス、キーボード等である。

【0009】本願の第1の発明によれば、動画像データ保持手段は、動画像データを保持する。表示手段は、前記動画像データに基づく動画像を表示する。動体領域抽出手段は、前記動画像データのフレーム中から動体部分の画像データである動体領域を抽出する。特微量データ抽出手段は、前記動体領域抽出手段によって抽出された動体領域を特徴づける特微量データを抽出する。特微量データベース手段は、前記特微量データに対応させて前記動体領域を含むフレームを特定するフレーム特定データを保持する。特微量データ入力手段によって、検索対象の動画像データの前記特微量データが入力される。特微量データベース検索手段は、前記特微量データ入力手段によって入力された前記特微量データに基づいて前記特微量データベース手段を検索して前記フレーム特定データを得る。制御手段は、前記特微量データベース検索手段によって得られた前記フレーム特定データによって特定されるフレームを前記動画像データ保持手段から読み出して前記表示手段に表示させる。

【0010】このように、動画像データから動体領域を特徴づける特微量データが抽出され、当該特微量データに基づいて動画像検索がなされるので、動体領域以外の背景の影響を受けることがない。

【0011】また、本願の第2の発明は、上述した課題を解決するため、前記動画像データはMPEGデータであり、前記動体領域抽出手段は、MPEGデータ中の動ベクトルの長さが所定の閾値以上であるマクロブロックが連接した領域を前記動体領域として抽出することとして、請求項1記載の発明を特定したものである（請求項2に対応）。

【0012】また、本願の第3の発明は、上述した課題を解決するため、前記動画像データはMPEGデータであり、前記動体領域抽出手段は、MPEGデータ中の動ベクトルの長さが所定の閾値以上であるマクロブロックが連接した領域と、MPEGデータ中のPフレーム及びBフレーム内のIntra-codedマクロブロックが連接した領域とを、前記動体領域として抽出することとして、請求項1記載の発明を特定したものである（請求項3に対応）。

【0013】また、本願の第4の発明は、上述した課題を解決するため、動画像データを保持する動画像データ保持手段と、前記動画像データに基づく動画像を表示する表示手段と、前記動画像データ中のある時点における画像データを背景データとして保持する背景データ保持手段と、前記時点以降のフレーム中の領域であって前記背景データとの色彩の差が所定の閾値以上である領域を動体領域として抽出する動体領域抽出手段と、前記動体

領域抽出手段によって抽出された動体領域を特徴づける特微量データを抽出する特微量データ抽出手段と、前記特微量データに対応させて前記動体領域を含むフレームを特定するフレーム特定データを保持する特微量データベース手段と、検索対象の動画像データの前記特微量データを入力するための特微量データ入力手段と、前記特微量データ入力手段によって入力された前記特微量データに基づいて前記特微量データベース手段を検索して前記フレーム特定データを得る特微量データベース検索手段と、前記特微量データベース検索手段によって得られた前記フレーム特定データによって特定されるフレームを前記動画像データ保持手段から読み出して前記表示手段に表示させる制御手段とを備えたことを特徴とする（請求項4に対応）。

【0014】ここで、動画像データ保持手段、特微量データベース手段、背景データ保持手段とはハードディスク等であり、表示手段とはCRT等であり、動体領域抽出手段、特微量データ抽出手段、特微量データベース検索手段、制御手段とはCPU、DSP等であり、特微量データ入力手段とはマウス、キーボード等である。

【0015】本願の第4の発明によれば、動画像データ保持手段は、動画像データを保持する。表示手段は、前記動画像データに基づく動画像を表示する。背景データ保持手段は、前記動画像データ中のある時点における画像データを背景データとして保持する。動体領域抽出手段は、前記時点以降の前記動画像データ中の領域であって前記背景データとの色彩の差が所定の閾値以上である領域を動体領域として抽出する。特微量データ抽出手段は、前記動体領域抽出手段によって抽出された動体領域を特徴づける特微量データを抽出する。特微量データベース手段は、前記特微量データに対応させて前記動体領域を含むフレームを特定するフレーム特定データを保持する。特微量データ入力手段によって、検索対象の動画像データの前記特微量データが入力される。特微量データベース検索手段は、前記特微量データ入力手段によって入力された前記特微量データに基づいて前記特微量データベース手段を検索して前記フレーム特定データを得る。制御手段は、前記特微量データベース検索手段によって得られた前記フレーム特定データによって特定されるフレームを前記動画像データ保持手段から読み出して前記表示手段に表示させる。

【0016】このように、動画像データから動体領域を特徴づける特微量データが抽出され、当該特微量データに基づいて動画像検索がなされるので、動体領域以外の背景の影響を受けることがない。

【0017】また、本願の第5の発明は、上述した課題を解決するため、前記動画像データはMPEGデータであり、前記背景データ保持手段は、MPEGデータ中の各ショット内における最初のIフレーム中のマクロブロックのDC成分のデータを前記背景データとして保持す

るものであり、前記動体領域抽出手段は、前記背景データ保持手段に保持された前記Iフレーム以降の各Iフレーム中のマクロブロックのDC成分と前記背景データ中のデータであって前記マクロブロックの位置に対応する位置のデータとの差を算出し、その差が所定の閾値以上である前記マクロブロックが連接する領域を前記動体領域として抽出するものであることとして、請求項4記載の発明を特定したものである（請求項5に対応）。

【0018】また、本願の第6の発明は、上述した課題を解決するため、MPEGデータを保持する動画像データ保持手段と、MPEGデータに基づく動画像を表示する表示手段と、MPEGデータ中の各ショット内における最初のIフレーム中のマクロブロックのDC成分のデータを背景データとして保持する背景データ保持手段と、前記背景データ保持手段に保持された前記Iフレーム以降の各Iフレーム中のマクロブロックのDC成分と前記背景データ中のデータであって前記マクロブロックの位置に対応する位置のデータとの差を算出し、その差が所定の閾値以上である前記マクロブロックが連接する領域を前記動体領域として抽出するとともに、前記背景データ保持手段に保持された前記Iフレーム以降の各Pフレーム及びBフレーム中の動ベクトルの長さが所定の閾値以上であるマクロブロック及びIntra-codedマクロブロックが連接した領域を前記動体領域として抽出する動体領域抽出手段と、前記動体領域抽出手段によって抽出された動体領域を特徴づける特微量データを抽出する特微量データ抽出手段と、前記特微量データに対応させて前記動体領域を含むフレームを特定するフレーム特定データを保持する特微量データベース手段と、検索対象の動画像データの前記特微量データを入力するための特微量データ入力手段と、前記特微量データ入力手段によって入力された前記特微量データに基づいて前記特微量データベース手段を検索して前記フレーム特定データを得る特微量データベース検索手段と、前記特微量データベース検索手段によって得られた前記フレーム特定データによって特定されるフレームを前記動画像データ保持手段から読み出して前記表示手段に表示させる制御手段とを備えたことを特徴とする（請求項6に対応）。

【0019】ここで、動画像データ保持手段、特微量データベース手段、背景データ保持手段とはハードディスク等であり、表示手段とはCRT等であり、動体領域抽出手段、特微量データ抽出手段、特微量データベース検索手段、制御手段とはCPU、DSP等であり、特微量データ入力手段とはマウス、キーボード等である。

【0020】本願の第6の発明によれば、動画像データ保持手段は、MPEGデータを保持する。表示手段は、MPEGデータに基づく動画像を表示する。背景データ保持手段は、MPEGデータ中の各ショット内における最初のIフレーム中のマクロブロックのDC成分のデータを背景データとして保持する。

8

タを背景データとして保持する。動体領域抽出手段は、前記背景データ保持手段に保持された前記Iフレーム以降の各Iフレーム中のマクロブロックのDC成分と前記背景データ中のデータであって前記マクロブロックの位置に対応する位置のデータとの差を算出し、その差が所定の閾値以上である前記マクロブロックが連接する領域を前記動体領域として抽出するとともに、前記背景データ保持手段に保持された前記Iフレーム以降の各Pフレーム及びBフレーム中の動ベクトルの長さが所定の閾値以上であるマクロブロック及びIntra-codedマクロブロックが連接した領域を前記動体領域として抽出する。特微量データ抽出手段は、前記動体領域抽出手段によって抽出された動体領域を特徴づける特微量データを抽出する。特微量データベースは、前記特微量データに対応させて前記動体領域を含むフレームを特定するフレーム特定データを保持する。特微量データ入力手段によって、検索対象の動画像データの前記特微量データが入力される。特微量データベース検索手段は、前記特微量データ入力手段によって入力された前記特微量データに基づいて前記特微量データベース手段を検索して前記フレーム特定データを得る。制御手段は、前記特微量データベース検索手段によって得られた前記フレーム特定データによって特定されるフレームを前記動画像データ保持手段から読み出して前記表示手段に表示させる。

【0021】このように、MPEGデータから動体領域を特徴づける特微量データが抽出され、当該特微量データに基づいて動画像検索がなされるので、動体領域以外の背景の影響を受けることがない。

【0022】また、本願の第7の発明は、上述した課題を解決するため、前記特微量データ抽出手段は、所定数の色彩を基準に前記動体領域を分割して1つ以上の色領域とし、前記色領域の大きさ及び色彩を前記特微量データとするものであり、前記特微量データ入力手段は、検索対象の大きさ及び色彩を入力するものであり、前記特微量データベース検索手段は、前記特微量データ入力手段によって入力された検索対象の大きさ及び色彩によって前記特微量データベース手段を検索するものであることとして、請求項1乃至6の何れかに記載の発明を特定したものである（請求項7に対応）。

【0023】また、本願の第8の発明は、上述した課題を解決するため、前記特微量データ抽出手段は、前記色領域の大きさ及び色彩に加えて、前記動体領域の移動方向ベクトルを前記特微量データとするものであり、前記特微量データ入力手段は、検索対象の大きさ、色彩及び移動方向ベクトルを入力するものであり、前記特微量データベース検索手段は、前記特微量データ入力手段によって入力された検索対象の大きさ、色彩及び移動方向ベクトルによって前記特微量データベース手段を検索するものであることとして、請求項7記載の発明を特定したものである（請求項8に対応）。

【0024】

【発明の実施の形態】以下、本発明の実施の形態を図1乃至図43の図面に基いて説明する。

【0025】

【第1実施形態】図1は本発明の実施形態による動画像検索システム10の回路構成を示したものである。図1において、本実施形態による動画像検索システム10は、マウス11、キーボード12、表示装置13、コンピュータ本体20から、構成されている。またコンピュータ本体20は、バスBによって相互に接続されたマウスインターフェース21、キーボードインターフェース22、表示装置インターフェース23、画像インターフェース24、記憶装置インターフェース26、メモリ27、制御部28と、記憶装置インターフェース26を介してバスBと接続された記憶装置25とから、構成されている。マウス11はマウスインターフェース21を介して、キーボード12はキーボードインターフェース22を介して、表示装置13は表示装置インターフェース23を介して、それぞれバスBと接続されている。

【0026】マウス11は、ユーザがデータを入力するための装置である。キーボード12は、ユーザが文字等のデータを入力するための入力装置である。このマウス11及びキーボード12が、特徴量データ入力手段に相当する。

【0027】表示装置13は、キーボード12から入力されたデータの文字、画像等を表示するための装置である。この表示装置13が、表示手段に相当する。コンピュータ本体20は、プログラムの実行等を行うための装置である。

【0028】マウスインターフェース21は、ユーザから入力されたデータをマウス11から受け取ってバスBへ伝達する装置である。キーボードインターフェース22は、キーボード12から入力データを受け取ってバスBへ伝達する装置である。

【0029】表示装置インターフェース23は、文字、画像等のデータを表示装置13に表示させるための装置である。画像インターフェース24は、画像伝送路Cを介してMPEG(Moving Picture Experts Group)データの受信を行う装置である。

【0030】記憶装置25は、MPEGデータ、制御部28が処理するプログラム、特徴量データベース、カット情報データベース等を記憶するハードディスク装置である。ここで、特徴量データベースとは、MPEGデータのフレーム中の動体部分の画像データの移動方向ベクトル、色彩、大きさ(面積)、及び重心座標からなる特徴量データに対応させて、MPEGデータ中の当該動体領域を含むフレーム番号を記憶するデータベースである。また、カット情報データベースとは、MPEGデータ中のカット番号に対応させて、MPEGデータ中のカット位置のフレーム番号を記憶するデータベースであ

る。この記憶装置25が、動画像データ保持手段、及び特徴量データベース手段に相当する。

【0031】記憶装置インターフェース26は、記憶装置25へのデータの書き込みや、記憶装置25からのデータの読み出しを行う装置である。メモリ27は、RAM等で構成され、制御部28の作業用に用いられるメモリである。このメモリ27が、背景データ保持手段に相当する。

【0032】制御部28は、CPU(Central

10 Processing Unit), DSP(Digital Signal Processor)等で構成され、表示装置インターフェース23に対して、画面表示指示を行う。また、制御部28は、記憶装置インターフェース26に対して、記憶装置25へのデータ書き込み指示、記憶装置25からのデータ読み出し指示を行う。更に、制御部28は、マウスインターフェース21を介してマウス11から入力データを受け取り、キーボードインターフェース22を介してキーボード12から入力データを受け取り、画像インターフェース24を介してMPEG画像データを受け取る。また、制御部28は、マウス11及びキーボード12から入力された文字等のデータの処理、表示装置13に表示する画面データの処理を行う。この制御部28が、動体領域抽出手段、特徴量データ抽出手段、特徴量データベース検索手段、及び制御手段に相当する。

<第1実施形態の動作の説明>次に、記憶装置25に格納されているとともに制御部28で実行される動画像検索プログラムの内容を、図2乃至図17のフローチャートによって説明する。

30 <MPEGデータの取り込み及び特徴量データ抽出処理>動画像検索システム10の制御部28は、オペレータによってキーボード12から画像データ取り込み指示データが入力されると、図2乃至図13の処理を実行する。

【0033】図2乃至図13の処理を開始すると、最初のS001では、動画像検索システム10の制御部28は、画像インターフェース24を介してMPEGデータを受け取り、記憶装置25に書き込む。

【0034】次のS002では、制御部28は、S00401で受け取ったMPEGデータ中の全てのカット位置を検出し、記憶装置25中のカット情報データベースに書き込む。

【0035】次のS003では、制御部28は、S002で検出したカット位置のデータからショット数を求め、変数Snumに代入する。次のS004では、制御部28は、各フレーム中のマクロブロックの数を求め、変数MBnumに代入する。

【0036】次のS005では、制御部28は、変数s,fを0に初期化する。次に処理は、S006乃至S50 099のループに入る。このループに入つて最初のS0

06では、制御部28は、 s を1つインクリメントする。

【0037】次のS007では、制御部28は、第sショット中のフレーム数を求め、変数Fnumに代入する。次のS008では、制御部28は、配列型変数 $p_{vx}[1..MBnum]$, $p_{vy}[1..MBnum]$ を0に初期化する。この p_{vx} は、各マクロブロックの予測動ベクトルのx成分を保持する変数である。同様に、 p_{vy} は、各マクロブロックの予測動ベクトルのy成分を保持する変数である。

【0038】次に処理は、S009乃至S098のループに入る。このループに入って最初のS009では、制御部28は、 f を1つインクリメントする。

【0039】次のS010では、制御部28は、記憶装置25から第sショットの第fフレーム中の各マクロブロックの動ベクトルを読み出し、そのx成分を配列型変数 $v_{x}[1..MBnum]$ に、y成分を $v_{y}[1..MBnum]$ に夫々代入する。

【0040】次のS011では、制御部28は、第fフレームがI(Itra)フレームであるか否かをチェックする。そして、もしIフレームであれば処理をS012へ移し、そうでなければ(即ちP(Predictive)フレーム又はB(Bidirectional predictiv)フレームであれば)処理をS013へ移す。

【0041】S012では、制御部28は、動ベクトルに代えて予測動ベクトルを使用するべく、配列型変数 p_{vx} の各要素の値を v_x の各要素に、 p_{vy} の各要素の値を v_y の各要素に夫々代入する。

【0042】S013では、制御部28は、配列型変数 $r_k[1..MBnum]$ を0に初期化する。この配列型変数 r_k は、各マクロブロックが表す画像が動体の画像であるか否かを示す変数である。即ち、あるマクロブロックが表す画像が動体の画像である場合には、当該マクロブロックの番号を添字とする r_k の要素に1が書き込まれる。また、あるマクロブロックが表す画像が動体の画像でない場合には、当該マクロブロックの番号を添字とする r_k の要素に0が書き込まれる。

【0043】次に処理は、S014乃至S019のループに入る。このループに入って最初のS014では、制御部28は、 m を1つインクリメントする。

【0044】次のS015では、制御部28は、第sショットの第fフレーム中の第mマクロブロックがIntra-codedマクロブロックであるか否かをチェックする。そして、もしIntra-codedマクロブロックであれば処理をS018へ移し、そうでなければ処理をS016へ移す。

【0045】S016では、制御部28は、 $v_x[m]$ 及び $v_y[m]$ のデータからベクトル長を算出する。次のS017では、制御部28は、S016で算出したベ

クトル長が所定の閾値以上であるか否かをチェックする。そして、もし閾値以上であれば処理をS018へ移し、そうでなければ処理をS019へ移す。

【0046】S015にて第mマクロブロックがIntra-codedマクロブロックであると判断した場合又はS017にてベクトル長が閾値以上であると判断した場合には、次のS018において、制御部28は、 $r_k[m]$ に1を代入する。その後、処理はS019へ進む。

10 【0047】S019では、制御部28は、 m の値がMBnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS014へ戻す。以上説明したS014乃至S019のループを繰り返した結果、 m がMBnumと等しくなった場合には処理はS019からこのループを抜けて、S020に進む。

【0048】次のS020では、制御部28は、変数mを0に初期化する。次に処理は、S021乃至S026のループに入る。このループに入って最初のS021では、制御部28は、 m を1つインクリメントする。

20 【0049】次のS022では、制御部28は、 $r_k[m]$ が1であるか否かをチェックする。そして、もし1であれば処理をS023へ移し、そうでなければ処理をS026へ移す。

【0050】S023では、制御部28は、第mマクロブロックを囲む8個のマクロブロックを求める。次のS024では、制御部28は、配列型変数 r_k の要素のうちS023で求めた8個のマクロブロックに対応する要素の値が全て0であるか否かを、チェックする。そして、もし全て0であれば処理をS025へ移し、そうでなければ処理をS026へ移す。

【0051】S025では、制御部28は、 $r_k[m]$ に0を代入する。その後、処理はS026へ進む。S026では、制御部28は、 m の値がMBnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS021へ戻す。

【0052】以上説明したS021乃至S026のループを繰り返した結果、 m がMBnumと等しくなった場合には処理はS026からこのループを抜けて、S027に進む。

40 【0053】次のS027では、制御部28は、変数mを0に初期化する。次に処理は、S028乃至S033のループに入る。このループに入って最初のS028では、制御部28は、 m を1つインクリメントする。

【0054】次のS029では、制御部28は、 $r_k[m]$ が1であるか否かをチェックする。そして、もし1でなければ処理をS030へ移し、そうでなければ処理をS033へ移す。

【0055】S030では、制御部28は、配列型変数 r_k の要素であって画面上第mマクロブロックの左上、上、及び右上に位置する3個のマクロブロックに対応す

るものうち2つ以上が1であり、且つ、 r_k の要素であって画面上第mマクロブロックの左下、下、及び右下に位置する3個のマクロブロックに対応するもののうち2つ以上が1であるか否かをチェックする。そして、もし以上の条件を満たさない場合には処理をS031へ移し、満たす場合には処理をS032へ移す。

【0056】S031では、制御部28は、配列型変数 r_k の要素であって画面上第mマクロブロックの左上、左、及び左下に位置する3個のマクロブロックに対応するもののうち2つ以上が1であり、且つ、 r_k の要素であって画面上第mマクロブロックの右上、右、及び右下に位置する3個のマクロブロックに対応するもののうち2つ以上が1であるか否かをチェックする。そして、もし以上の条件を満たす場合には処理をS032へ移し、満たさない場合には処理をS033へ移す。

【0057】S032では、制御部28は、 $r_k[m]$ に1を代入する。その後、処理はS033へ進む。次のS033では、制御部28は、mの値がMBnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS028へ戻す。

【0058】以上説明したS028乃至S033のループを繰り返した結果、mがMBnumと等しくなった場合には処理はS033からこのループを抜けて、S034に進む。

【0059】次のS034では、制御部28は、変数mを0に初期化する。次に処理は、S035乃至S041のループに入る。このループに入って最初のS035では、制御部28は、mを1つインクリメントする。

【0060】次のS036では、制御部28は、第fフレーム中の第mマクロブロックの画面上のx座標を変数mxに、y座標を変数myに夫々代入する。次のS037では、制御部28は、マクロブロックの幅を変数mwに代入する。

【0061】次のS038では、制御部28は、x座標が $mx + vx[m] / mw$ 、y座標が $my + vy[m] / mw$ であるマクロブロックを第fフレーム中から抽出する。

【0062】次のS039では、制御部28は、S038で抽出したマクロブロックの番号を変数nに代入する。次のS040では、制御部28は、 $vx[n] + vx[m] / DAMP$ を計算して予測動ベクトルのx成分とし、 $pvx[n]$ に代入する。同様に、 $vy[n] + vy[m] / DAMP$ を計算して予測動ベクトルのy成分とし、 $pvy[n]$ に代入する。ここでDAMPは、所定の定数である。

【0063】次のS041では、制御部28は、mの値がMBnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS035へ戻す。以上説明したS035乃至S041のループを繰り返した結果、mがMBnumと等しくなった場合には処理はS041か

らこのループを抜けて、S042に進む。

【0064】次のS042では、制御部28は、配列型変数 r_k の要素の値が1であるものに対応するマクロブロックが連接する領域を動体領域候補として検出し、夫々に仮のラベル番号を付与する。

【0065】次のS043では、制御部28は、S042で検出した動体領域候補の数を変数DRKn umに代入する。次のS044では、制御部28は、nを0に初期化する。

10 【0066】次に処理は、S045乃至S057のループに入る。このループに入って最初のS045では、制御部28は、nを1つインクリメントする。

【0067】次のS046では、制御部28は、第n動体領域候補中のマクロブロックの数を変数MBDRKn umに代入する。次のS047では、制御部28は、MBDRKn umの値が所定の閾値以下であるか否かをチェックする。そして、閾値以下であれば処理をS048に移し、そうでなければ処理をS049に移す。

【0068】S048では、制御部28は、第n動体領域候補を構成する各マクロブロックに対応する r_k の要素に0を代入する。そして、処理はその後S057に進む。これに対し、S047にてMBDRKn umの値が所定の閾値以下でないと判断した場合には、次のS049において、制御部28は、第fフレーム中において第n動体領域候補が占める範囲に相当する第(f-1)フレーム中の範囲に、動体領域があるか否かをチェックする。そして、もしそのような動体領域がなければ処理をS050へ移し、そのような動体領域があれば処理をS051へ移す。

30 【0069】S050では、制御部28は、第n動体領域候補に新たなラベル番号を付与する。そして、そのラベル番号を確定したラベル番号とする。その後、処理はS052へ進む。

【0070】これに対し、S049にて動体領域があると判断した場合には、次のS051において、制御部28は、S049で見つけ出した動体領域のうち第n動体領域候補が占める範囲に相当する範囲に最も重なっている動体領域を選択する。そして、当該動体領域に付与されたラベル番号を第n動体領域候補に付与し、確定した40 ラベル番号とする。その後、処理はS052へ進む。

【0071】次のS052では、制御部28は、変数ax, ay, c, mを0に初期化する。次に処理は、S053乃至S055のループに入る。

【0072】このループに入って最初のS053では、制御部28は、mを1つインクリメントする。次のS054では、制御部28は、axにvx[m]を加算し、ayにvy[m]を加算する。更に、cをインクリメントする。

【0073】次のS055では、制御部28は、mがMBDRKn umと等しいか否かをチェックする。そし

て、もし等しくなければ処理を S053 へ戻す。以上説明した S053 乃至 S055 のループを繰り返した結果、m が M B D R K n u m と等しくなった場合には処理は S055 からこのループを抜けて、S056 に進む。

【0074】次の S056 では、制御部 28 は、第 n 動体領域候補の移動方向の x 成分を a_x / c 、y 成分を a_y / c として算出する。そして、このように算出された a_x / c 及び a_y / c の値を、S050 又は S051 で第 n 動体領域候補に付与されたラベル番号に対応させて、記憶装置 25 中の特徴量データベースに書き込む。

【0075】次の S057 では、制御部 28 は、n が D R K n u m と等しいか否かをチェックする。そして、もし等しくなければ処理を S045 へ戻す。以上説明した S045 乃至 S057 のループを繰り返した結果、n が DRKnum と等しくなった場合には、処理は S057 からこのループを抜けて S058 に進む。

【0076】次の S058 では、制御部 28 は、第 f フレームのデータを Y C b C r 表色系の画像データに復号化する。次の S059 では、制御部 28 は、S058 で得た Y C b C r 表色系画像データを L * a * b * 表色系に基づく画像データに変換する。次の S060 では、制御部 28 は、r_k の要素の値が 1 であるものに対応するマクロブロックが連接する領域を動体領域として検出し、その数を変数 R n u m に代入する。

【0077】次の S061 では、制御部 28 は、n を 0 に初期化する。次に、処理は、S062 乃至 S097 のループに入る。このループに入って最初の S062 では、制御部 28 は、n をインクリメントする。

【0078】次の S063 では、制御部 28 は、所定数 (C 個) の L * a * b * 表色系に基づく代表色の初期値を乱数によって夫々生成する。そして、これらの代表色の L * を配列型変数 l [1..C] に、a * を a [1..C] に、b * を b [1..C] に夫々代入する。

【0079】次の S064 では、制御部 28 は、変数 i を 0 に初期化する。次に、処理は、S065 乃至 S083 のループに入る。このループに入って最初の S065 では、制御部 28 は、i をインクリメントする。

【0080】次の S066 では、制御部 28 は、第 n 動体領域内の画素の数を変数 G n u m に代入する。次の S067 では、制御部 28 は、変数 g を 0 に初期化する。

【0081】次に、処理は、S068 乃至 S070 のループに入る。このループに入って最初の S068 では、制御部 28 は、g をインクリメントする。

【0082】次の S069 では、制御部 28 は、第 g 画素の色に最も近い色を C 個の代表色の中から選択し、その番号を配列型変数 d s [g] に代入する。尚、色の近さは、L *、a *、及び b * の値によって張られる 3 次元空間におけるユークリッド距離の大小によって判断する。以下において色の近さを判断するときも、同様とす

る。

【0083】次の S070 では、制御部 28 は、g が G n u m と等しいか否かをチェックする。そして、もし等しくなければ処理を S068 へ戻す。以上説明した S068 乃至 S070 のループを繰り返した結果、g が G n u m と等しくなった場合には、処理は S070 からこのループを抜けて S071 に進む。

【0084】次の S071 では、制御部 28 は、変数 c を 0 に初期化する。次に、処理は、S072 乃至 S081 のループに入る。このループに入って最初の S072 では、制御部 28 は、c をインクリメントする。

【0085】次の S073 では、制御部 28 は、変数 g, c1, ca, cb, ch を 0 に初期化する。次に、処理は、S074 乃至 S078 のループに入る。

【0086】このループに入って最初の S074 では、制御部 28 は、g をインクリメントする。次の S075 では、制御部 28 は、d s [g] の値が c と等しいか否かをチェックする。そして、もし等しければ処理を S076 へ移し、そうでなければ処理を S078 へ移す。

【0087】S076 では、制御部 28 は、第 g 画素の a * を変数 ga に、b * を gb に、L * を g1 に夫々代入する。次の S077 では、制御部 28 は、c1 に g1 を加算し、ca に ga を加算し、cb に gb を加算する。更に、ch をインクリメントする。

【0088】次の S078 では、制御部 28 は、g が G n u m と等しいか否かをチェックする。そして、もし等しくなければ処理を S074 へ戻す。以上説明した S074 乃至 S078 のループを繰り返した結果、g が G n u m と等しくなった場合には、処理は S078 からこのループを抜けて S079 に進む。

【0089】次の S079 では、制御部 28 は、c1 / ch を算出して l [c] に代入し、ca / ch を算出して a [c] に代入し、cb / ch を算出して b [c] に代入する。更に、h [c] に ch を代入する。

【0090】次の S080 では、制御部 28 は、c が C と等しいか否かをチェックする。そして、もし等しくなければ処理を S072 へ戻す。以上説明した S072 乃至 S080 のループを繰り返した結果、c が C と等しくなった場合には、処理は S080 からこのループを抜けて S081 に進む。

【0091】次の S081 では、制御部 28 は、i が所定の繰り返し回数の値に達したか否かをチェックする。そして、もし達していないければ処理を S082 へ移し、達していれば S065 乃至 S081 のループを抜けて処理を S084 へ移す。

【0092】S082 では、制御部 28 は、配列型変数 h の要素の最大値の添字を変数 max に代入する。また、最小値の添字を変数 min に代入する。次の S083 では、制御部 28 は、l [max] の値に乱数を加算した値を l [min] に代入する。同様に、a [ma

x] の値に乱数を加算した値を a [m i n] に、 b [m a x] の値に乱数を加算した値を b [m i n] に夫々代入する。尚、乱数の値は、 l [m a x] , a [m a x] , 及び b [m a x] の値に比して小さな値とする。その後、処理は S 0 6 5 へ戻る。

【0093】一方、 S 0 8 1 にて i が所定の繰り返し回数の値に達したと判断した場合、次の S 0 8 4 において、制御部 2 8 は、変数 g を 0 に初期化する。次に、処理は、 S 0 8 5 乃至 S 0 8 7 のループに入る。

【0094】このループに入って最初の S 0 8 5 では、制御部 2 8 は、 g をインクリメントする。次の S 0 8 6 では、制御部 2 8 は、第 g 画素の色に最も近い色を C 個の代表色の中から選択し、その代表色の番号を配列型変数 $d s$ [g] に代入する。

【0095】次の S 0 8 7 では、制御部 2 8 は、 g が G n u m と等しいか否かをチェックする。そして、もし等しくなければ処理を S 0 8 5 に戻す。以上説明した S 0 8 5 乃至 S 0 8 7 のループを繰り返した結果、 g が G n u m と等しくなった場合には、処理は S 0 8 7 からこのループを抜けて S 0 8 8 に進む。

【0096】次の S 0 8 8 では、制御部 2 8 は、配列型変数 $d s$ の値が等しい要素に対応するマクロブロックが接続する領域を、色領域として検出する。次の S 0 8 9 では、制御部 2 8 は、 S 0 8 8 で検出した色領域の数を変数 I R n u m に代入する。

【0097】次の S 0 9 0 では、制御部 2 8 は、変数 k を 0 に初期化する。次に、処理は、 S 0 9 1 乃至 S 0 9 6 のループに入る。このループに入って最初の S 0 9 1 では、制御部 2 8 は、 k をインクリメントする。

【0098】次の S 0 9 2 では、制御部 2 8 は、ノイズのカット等のため、第 k 色領域を縮小する。次の S 0 9 3 では、制御部 2 8 は、第 k 色領域を拡大する。

【0099】次の S 0 9 4 では、制御部 2 8 は、第 k 色領域を構成する画素数が所定の閾値以上であるか否かをチェックする。そして、もし閾値以上であれば処理を S 0 9 5 へ移し、そうでなければ処理を S 0 9 6 へ移す。

【0100】S 0 9 5 では、制御部 2 8 は、第 k 色領域の色に最も近い代表色の番号、第 k 色領域の画素数、及び第 k 色領域の重心座標を、第 n 動体領域のラベル番号である n に対応させて、記憶装置 2 5 中の特徴量データベースに書き込む。

【0101】次の S 0 9 6 では、制御部 2 8 は、 k が I R n u m と等しいか否かをチェックする。そして、もし等しくなければ処理を S 0 9 1 に戻す。以上説明した S 0 9 1 乃至 S 0 9 6 のループを繰り返した結果、 k が I R n u m と等しくなった場合には、処理は S 0 9 6 からこのループを抜けて S 0 9 7 に進む。

【0102】次の S 0 9 7 では、制御部 2 8 は、 n が R n u m と等しいか否かをチェックする。そして、もし等しくなければ処理を S 0 6 2 に戻す。以上説明した S 0

6 2 乃至 S 0 9 7 のループを繰り返した結果、 n が R n u m と等しくなった場合には、処理は S 0 9 7 からこのループを抜けて S 0 9 8 に進む。

【0103】次の S 0 9 8 では、制御部 2 8 は、 f が F n u m と等しいか否かをチェックする。そして、もし等しくなければ処理を S 0 0 9 に戻す。以上説明した S 0 0 9 乃至 S 0 9 8 のループを繰り返した結果、 f が F n u m と等しくなった場合には、処理は S 0 9 8 からこのループを抜けて S 0 9 9 に進む。

10 【0104】次の S 0 9 9 では、制御部 2 8 は、 s が S n u m と等しいか否かをチェックする。そして、もし等しくなければ処理を S 0 0 6 に戻す。以上説明した S 0 0 6 乃至 S 0 9 9 のループを繰り返した結果、 s が S n u m と等しくなった場合には、処理は S 0 9 9 からこのループを抜けて処理を終了する。

【0105】以上のようにして、制御部 2 8 は、 M P E G データ中から、動体領域を抽出するとともに、各動体領域の移動方向ベクトルを算出することができる。また、各動体領域中の色領域を抽出するとともに、各色領域の色に最も近い代表色番号、各色領域の画素数、及び各色領域の重心座標を検出することができる。

<検索データの入力処理及び検索処理>次に、検索のための特徴量データの入力処理及び当該特徴量データに基づく検索処理について、図 1 4 乃至図 1 7 のフローチャートに基づいて説明する。

【0106】動画像検索システム 1 0 の制御部 2 8 は、オペレータによってキーボード 1 2 から検索データ入力開始指示データが入力されると、図 1 4 乃至図 1 7 の処理を実行する。

30 【0107】図 1 4 乃至図 1 7 の処理を開始すると、最初の S 1 0 1 では、動画像検索システム 1 0 の制御部 2 8 は、入力された検索データをマウス 1 1 及びキーボード 1 2 から受け取る。ここで、検索データは、大きさ(画素数)のデータ及び L * a * b * 表色系による色のデータからなる 1 つ以上の検索色領域データと、各色領域データにオプションとして付加される検索移動方向データとから、構成される。

【0108】次の S 1 0 2 では、制御部 2 8 は、 S 1 0 1 で受け取った検索データ中の各色領域データをその画素数の多い順にソートする。次の S 1 0 3 では、制御部 2 8 は、 S 1 0 2 でソートした検索色領域データの中から、上位 N 個を選択する。

【0109】次の S 1 0 4 では、制御部 2 8 は、記憶装置 2 5 中の特徴量データベースに保持されている特徴量データの個数を変数 K に代入する。次の S 1 0 5 では、制御部 2 8 は、配列型変数 s m i n [1 .. K] , 及び変数 k を 0 に初期化する。

【0110】次に、処理は、 S 1 0 6 乃至 S 1 2 8 のループに入る。このループに入って最初の S 1 0 6 では、50 制御部 2 8 は、 k をインクリメントする。

【0111】次のS107では、制御部28は、第k特徴量データを記憶装置25中の特徴量データベースから読み出す。次のS108では、制御部28は、S101にて受け取った検索データに検索移動方向データが含まれているか否かをチェックする。そして、もし含まれていれば処理をS109へ移し、そうでなければ処理をS111へ移す。

【0112】S109では、制御部28は、第k特徴量データ中の移動方向データによって定まる移動方向が検索移動方向データによって定まる移動方向の±D度以内であるか否かをチェックする。そして、もし±D度以内でなければ処理をS110へ移し、±D度以内であれば処理をS111へ移す。

【0113】S110では、制御部28は、 $s_{min}[k]$ に、 $s_{min}[k]$ が採り得る最大値を代入する。その後処理は、S128に移る。これに対し、S109にて±D度以内であると判断した場合又はS108にて検索データに検索移動方向データが含まれていないと判断した場合には、次のS111において、制御部28は、第k特徴量データ中の色領域データを、その画素数の大きい順にソートする。

【0114】次のS112では、制御部28は、S111にてソートした色領域データの中から、上位N個を選択する。次のS113では、制御部28は、検索データ中に検索色領域データが1つだけ含まれているか否かをチェックする。そして、もし1つだけであれば処理をS114へ移し、そうでなければ処理をS115へ移す。

【0115】S114では、制御部28は、検索色領域データ中の色データによって定まる色と、第k特徴量データに含まれる全ての色領域データ中の色データによって定まる色との間の色差を夫々算出する。色差は、 L^* , a^* , 及び b^* によって張られる3次元空間におけるユークリッド距離によって算出する。そして、算出された値の中の最小値を $s_{min}[k]$ に代入する。その後、処理は、S128へ進む。

【0116】S115では、制御部28は、変数iを0に初期化する。次に、処理は、S116乃至S126のループに入る。このループに入って最初のS116では、制御部28は、iをインクリメントする。

【0117】次のS117では、制御部28は、第k特徴量データ中の第i色領域データによって定まる色と、第1検索色領域データによって定まる色との間の色差を算出し、 $s[i]$ に代入する。

【0118】次のS118では、制御部28は、変数nを0に初期化する。次に、処理は、S119至S125のループに入る。このループに入って最初のS119では、制御部28は、nをインクリメントする。

【0119】次のS120では、制御部28は、第1検索色領域データによって定まる領域の重心から第n色領域データ中の重心データによって定まる重心を見た時

の、所定の基準方向に対する角度を算出し、変数o_aに代入する。

【0120】次のS121では、制御部28は、第k特徴量データに含まれる第i色領域データによって定まる領域の重心から $o_a \pm D$ 度の角度範囲内にある色領域データを抽出する。

【0121】次のS122では、制御部28は、S121にて色領域データが抽出されたか否かをチェックする。そして、もし抽出されなければ処理をS123へ移し、そうでなければ処理をS124へ移す。

【0122】S123では、制御部28は、S121にて抽出された色領域データであってその色が第n検索色領域データによって定まる色と最も近い色領域データを、選択する。そして、このようにして選択された色領域データによって定まる色と第n検索色領域データによって定まる色との色差を算出し、 $s[i]$ に加算する。その後、処理はS125へ進む。

【0123】これに対し、S122にて色領域データが抽出されなかつたと判断した場合には、次のS124において、制御部28は、 $s[i]$ に、 $s[i]$ が採りうる最大値を代入する。その後処理は、S125に進む。

【0124】S125では、制御部28は、nがNと等しいか否かをチェックする。そして、もし等しくなければ処理をS119へ戻す。以上説明したS119乃至S125のループを繰り返した結果、nがNと等しくなつた場合には処理はS125からこのループを抜けて、S126に進む。

【0125】S126では、制御部28は、iがNと等しいか否かをチェックする。そして、もし等しくなければ処理をS116へ戻す。以上説明したS116乃至S126のループを繰り返した結果、iがNと等しくなつた場合には処理はS126からこのループを抜けて、S127に進む。

【0126】次のS127では、制御部28は、 $s[1..N]$ の要素の中の最小値を $s_{min}[k]$ に代入する。次のS128では、制御部28は、kがKと等しいか否かをチェックする。そして、もし等しくなければ処理をS106へ戻す。

【0127】以上説明したS106乃至S128のループを繰り返した結果、kがKと等しくなつた場合には処理はS128からこのループを抜けて、S129に進む。S129では、制御部28は、 $s_{min}[1..K]$ を値の小さい順にソートする。

【0128】次のS130では、制御部28は、S129でソートされた要素の中から上位M個を選択する。次のS131では、制御部28は、変数mを0に初期化する。

【0129】次に、処理は、S132至S134のループに入る。このループに入って最初のS132では、制御部28は、mをインクリメントする。

【0130】次のS133では、制御部28は、第m特徴量データによって特定される動体を含むフレームを、記憶装置25に記憶されたMPEGデータ中から抽出する。そして、当該フレームに基づく画像を表示装置13に表示させるように、表示装置コントローラ23に指示する。

【0131】次のS134では、制御部28は、mがMと等しいか否かをチェックする。そして、もし等しくなければ処理をS132へ戻す。以上説明したS132乃至S134のループを繰り返した結果、mがMと等しくなった場合には処理はS134からこのループを抜けて、処理を終了する。

【0132】以上のように、制御部28は、オペレータから入力された検索データに基づいて特徴量データベースを検索し、検索対象の動体を含むフレームの画像を表示装置13に表示させることができる。

【0133】

【第2実施形態】次に、本発明の第2の実施形態について説明する。第2の実施形態の回路構成は、先に説明した第1の実施形態と同様であり、図1によって示される。

【0134】次に、第2の実施形態の動作について、図18乃至図27のフローチャートによって説明する。
< MPEGデータの取り込み及び特徴量データ抽出処理 > 動画像検索システム10の制御部28は、オペレータによってキーボード12からMPEGデータ取り込み指示データが入力されると、図18乃至図27の処理を実行する。

【0135】図18乃至図27の処理を開始すると、最初のS201では、動画像検索システム10の制御部28は、画像インターフェース24を介してMPEGデータを受け取り、記憶装置25に書き込む。

【0136】次のS202では、制御部28は、S201で受け取ったMPEGデータ中の全てのカット位置を検出し、記憶装置25中のカット情報データベースに書き込む。

【0137】次のS203では、制御部28は、S202で検出したカット位置のデータからショット数を求め、変数Snumに代入する。次のS204では、制御部28は、各フレーム中のマクロブロックの数を変数MBnumに、ブロックの数を変数Bnumに夫々代入する。

【0138】次のS205では、制御部28は、変数s、fを0に初期化する。次に処理は、S206乃至S284のループに入る。このループに入つて最初のS206では、制御部28は、sを1つインクリメントする。

【0139】次のS207では、制御部28は、第sショット中のフレーム数を求め、変数Fnumに代入する。次に処理は、S208乃至S283のループに入

る。

【0140】このループに入つて最初のS208では、制御部28は、fを1つインクリメントする。次のS209では、制御部28は、第fフレームがI(Intra)フレームであるか否かをチェックする。そして、もしIフレームであれば処理をS210へ移し、そうでなければ処理をS283へ移す。

【0141】S210では、制御部28は、第fフレームが第sショットにおける最初のIフレームであるか否かをチェックする。そして、もし最初のIフレームであれば処理をS211へ移し、そうでなければ処理をS213へ移す。

【0142】S211では、制御部28は、第fフレーム中の各マクロブロックのDC成分データを読み出す。尚、各マクロブロック中には、Y(輝度信号)成分のためのブロックが4個ある。従って、Y成分のDC成分も4個あるので、4個の算術平均をY成分のDC成分とする。Cb及びCr(色相信号)成分のためのブロックは夫々1個があるので、夫々のDC成分をそのまま読み出す。

【0143】次のS212では、制御部28は、S211で読み出したDC成分データをL*a*b*表色系に変換し、dc1[1..MBnum]、dca[1..MBnum]、及びdcb[1..MBnum]に夫々代入する。

【0144】次のS213では、制御部28は、配列型変数rk[1..MBnum]を0に初期化する。次のS214では、制御部28は、変数mを0に初期化する。

【0145】次に処理は、S215乃至S220のループに入る。このループに入つて最初のS215では、制御部28は、mを1つインクリメントする。

【0146】次のS216では、制御部28は、第fフレーム中の第mマクロブロックのDC成分を読み出してL*a*b*表色系に変換し、その値をml、ma、及びmbに夫々代入する。

【0147】次のS217では、制御部28は、ml、ma、及びmbで定まる色とdc1[m]、dca[m]、及びdcb[m]で定まる色との色差を算出する。次のS218では、制御部28は、S217で算出した色差が所定の閾値以上であるか否かをチェックする。そして、もし閾値以上であれば処理をS219へ移し、そうでなければ処理をS220へ移す。

【0148】S219では、制御部28は、rk[m]に1を代入する。S220では、制御部28は、mがMBnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS215へ戻す。

【0149】以上説明したS215乃至S220のループを繰り返した結果、mがMBnumと等しくなった場合には処理はS220からこのループを抜けて、S22

1に進む。

【0150】次のS221では、制御部28は、変数mを0に初期化する。次に処理は、S222乃至S227のループに入る。このループに入って最初のS222では、制御部28は、mを1つインクリメントする。

【0151】次のS223では、制御部28は、r k [m]が1であるか否かをチェックする。そして、もし1であれば処理をS227へ移し、そうでなければ処理をS224へ移す。

【0152】S224では、制御部28は、第mマクロブロックを囲む8個のマクロブロックを求める。次のS225では、制御部28は、配列型変数r kの要素のうちS224で求めた8個のマクロブロックに対応するものの値が全て0であるか否かをチェックする。そして、もし全て0であれば処理をS226へ移し、そうでなければ処理をS227へ移す。

【0153】S226では、制御部28は、 $\{dcl[m] * (N-1) + m1\} / N$ を算出してdcl [m]に代入する。同様に、 $\{dca[m] * (N-1) + ma\} / N$ を算出してdca [m]に代入し、 $\{dcb[m] * (N-1) + mb\} / N$ を算出してdcb [m]に代入する。尚、第sショットの処理を開始してから現れたIフレームが未だN個に満たない場合には、上式のNをIフレームが現れた数によって置き換えて算出する。

【0154】次のS227では、制御部28は、mがMBnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS222へ戻す。以上説明したS222乃至S227のループを繰り返した結果、mがMBnumと等しくなった場合には処理はS227からこのループを抜けて、S228に進む。

【0155】次のS228では、制御部28は、配列型変数r kの要素のうち値が1であるものに対応するマクロブロックが連接する領域を動体領域候補として検出し、夫々に仮のラベル番号を付与する。

【0156】次のS229では、制御部28は、S228にて検出した動体領域候補の数を変数DRKnumに代入する。次のS230では、制御部28は、変数nを0に初期化する。

【0157】次に処理は、S231乃至S242のループに入る。このループに入って最初のS231では、制御部28は、nを1つインクリメントする。

【0158】次のS232では、制御部28は、第n動体領域候補中のマクロブロックの数を変数MBDRKnumに代入する。次のS233では、制御部28は、MBDRKnumの値が所定の閾値以下であるか否かをチェックする。そして、もし閾値以下であれば処理をS234へ移し、そうでなければ処理をS235へ移す。

【0159】S234では、制御部28は、第n動体領域候補を構成する各マクロブロックに対応するr kの要

素に0を代入する。その後、処理はS242へ進む。これに対し、S233にてMBDRKnumの値が所定の閾値以上であると判断した場合、次のS235において、制御部28は、第n動体領域候補の重心を算出する。そして、その重心のx座標を変数x_nに、y座標をy_nに夫々代入する。

【0160】次のS236では、制御部28は、第fフレームより1つ前のIフレーム中の範囲であって第fフレームにおける第n動体領域候補が占める範囲に相当する範囲に動体領域があるか否かをチェックする。そして、もしそのような動体領域がなければ処理をS237へ移し、あれば処理をS239へ移す。

【0161】S237では、制御部28は、第n動体領域候補に新しいラベル番号を付与する。そして、制御部28は、当該番号を第n動体領域候補の確定したラベル番号とする。

【0162】次のS238では、制御部28は、変数x_{n-1}にx_nの値を代入し、変数y_{n-1}にy_nの値を代入する。その後処理は、S241へ進む。これに対し、S236にて動体領域があると判断した場合、次のS239にて、制御部28は、S235にて検出した動体領域のうち第n動体領域候補が占める範囲に相当する範囲に重なるマクロブロック数が最も多い動体領域を選択する。そして、制御部28は、当該動体領域に付与されたマクロブロック番号を、第n動体領域候補に付与する。更に、制御部28は、当該番号を第n動体領域候補の確定したラベル番号とする。

【0163】次のS240では、制御部28は、S239にて選択した動体領域の重心座標を特徴量データベースから読み出す。そして、制御部28は、読み出した重心座標のx座標を変数x_{n-1}に、y座標を変数y_{n-1}に夫々代入する。その後、処理はS241に進む。

【0164】S241では、制御部28は、第n動体領域候補の移動方向のx成分をx_n - x_{n-1}によって算出し、y成分をy_n - y_{n-1}によって算出する。そして、第n動体領域候補のラベル番号であるnに対応させて、記憶装置25中の特徴量データベースに書き込む。動体領域候補は、このように特徴量データベースに書き込まれた時点で、動体領域となる。

【0165】次のS242では、制御部28は、nがRnumと等しいか否かをチェックする。そして、等しくなければ処理をS231へ戻す。以上説明したS231乃至S242のループを繰り返した結果、nがRnumと等しくなった場合には処理はS242からこのループを抜けて、S243に進む。

【0166】次のS243では、制御部28は、第fフレームのデータを画像データに復号化する。次のS244では、制御部28は、S243で得た画像データをL*a*b*表色系に基づく画像データに変換する。

【0167】次のS245では、制御部28は、配列型

変数 r_k の要素のうち値が 1 であるものに対応するマクロブロックが連接する動体領域を検出し、その数を変数 $Rnum$ に代入する。

【0168】次のS246では、制御部28は、変数nを0に初期化する。次に処理は、S247乃至S282のループに入る。このループに入って最初のS247では、制御部28は、nを1つインクリメントする。

【0169】次のS248では、制御部28は、所定数(C個)のL*a*b*表色系に基づく代表色の初期値を乱数によって夫々生成する。そして、これらの代表色のL*を配列型変数l [1..C]に、a*をa [1..C]に、b*をb [1..C]に夫々代入する。

【0170】次のS249では、制御部28は、変数iを0に初期化する。次に、処理は、S250乃至S268のループに入る。このループに入って最初のS250では、制御部28は、iをインクリメントする。

【0171】次のS251では、制御部28は、第n動体領域内の画素の数を変数Gnumに代入する。次のS252では、制御部28は、変数gを0に初期化する。

【0172】次に、処理は、S253乃至S255のループに入る。このループに入って最初のS253では、制御部28は、gをインクリメントする。

【0173】次のS254では、制御部28は、第g画素の色に最も近い色をC個の代表色の中から選択し、その番号を配列型変数ds [g]に代入する。尚、色の近さは、L*, a*, 及びb*の値によって張られる3次元空間におけるユークリッド距離の大小によって判断する。以下において色の近さを判断するときも、同様とする。

【0174】次のS255では、制御部28は、gがGnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS253に戻す。以上説明したS253乃至S255のループを繰り返した結果、gがGnumと等しくなった場合には、処理はS255からこのループを抜けてS256に進む。

【0175】次のS256では、制御部28は、変数cを0に初期化する。次に、処理は、S257乃至S265のループに入る。このループに入って最初のS257では、制御部28は、cをインクリメントする。

【0176】次のS258では、制御部28は、変数g, cl, ca, cb, chを0に初期化する。次に、処理は、S259乃至S263のループに入る。

【0177】このループに入って最初のS259では、制御部28は、gをインクリメントする。次のS260では、制御部28は、ds [g]がcと等しいか否かをチェックする。そして、もし等しければ処理をS261へ移し、等しくなければ処理をS263へ移す。

【0178】次のS261では、制御部28は、第g画素のa*を変数gaに、b*をgbに、L*をglに夫

々代入する。次のS262では、制御部28は、clにglを加算し、caにgaを加算し、cbにgbを加算する。更に、chをインクリメントする。

【0179】次のS263では、制御部28は、gがGnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS259に戻す。以上説明したS259乃至S263のループを繰り返した結果、gがGnumと等しくなった場合には、処理はS263からこのループを抜けてS264に進む。

10 【0180】次のS264では、制御部28は、cl/chを算出してl [c]に代入し、ca/chを算出してa [c]に代入し、cb/chを算出してb [c]に代入する。更に、h [c]にchを代入する。

【0181】次のS265では、制御部28は、cがCと等しいか否かをチェックする。そして、もし等しくなければ処理をS257に戻す。以上説明したS257乃至S265のループを繰り返した結果、cがCと等しくなった場合には、処理はS265からこのループを抜けてS266に進む。

20 【0182】次のS266では、制御部28は、iが所定の繰り返し回数の値に達したか否かをチェックする。そして、もし達していないければ処理をS267へ移し、そうでなければS250乃至S268のループを抜けて処理をS269へ移す。

【0183】S267では、制御部28は、配列型変数hの要素の最大値の添字を変数maxに代入する。また、最小値の添字を変数minに代入する。次のS268では、制御部28は、l [max]の値に乱数を加算した値をl [min]に代入する。同様に、a [max]の値に乱数を加算した値をa [min]に、b [max]

30 x]の値に乱数を加算した値をb [min]に夫々代入する。尚、乱数の値は、l [max], a [max], 及びb [max]の値に比して小さな値とする。その後、処理はS250へ戻る。

【0184】これに対し、S266にて所定の繰り返し回数に達したと判断した場合、次のS269において、制御部28は、変数gを0に初期化する。次に、処理は、S270乃至S272のループに入る。

【0185】このループに入って最初のS270では、40 制御部28は、gをインクリメントする。次のS271では、制御部28は、第g画素の色に最も近い色をC個の代表色の中から選択し、その代表色の番号を配列型変数ds [g]に代入する。

【0186】次のS272では、制御部28は、gがGnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS270に戻す。以上説明したS270乃至S272のループを繰り返した結果、gがGnumと等しくなった場合には、処理はS272からこのループを抜けてS273に進む。

50 【0187】次のS273では、制御部28は、配列型

変数 d_s の値が等しい要素に対応するマクロブロックが連接する領域を、色領域として検出する。次の S 274 では、制御部 28 は、S 272 で検出した色領域の数を変数 $IRnum$ に代入する。

【0188】次の S 275 では、制御部 28 は、変数 k を 0 に初期化する。次に、処理は、S 276 乃至 S 281 のループに入る。このループに入って最初の S 276 では、制御部 28 は、 k をインクリメントする。

【0189】次の S 277 では、制御部 28 は、ノイズのカット等のため、第 k 色領域を縮小する。次の S 278 では、制御部 28 は、第 k 色領域を拡大する。

【0190】次の S 279 では、制御部 28 は、第 k 色領域を構成する画素数が所定の閾値以上であるか否かをチェックする。そして、もし閾値以上であれば処理を S 280 へ移し、そうでなければ処理を S 281 へ移す。

【0191】S 280 では、制御部 28 は、第 k 色領域の色に最も近い代表色の番号、第 k 色領域の画素数、及び第 k 色領域の重心座標を、第 n 動体領域のラベル番号である n に対応させて、記憶装置 25 中の特徴量データベースに書き込む。

【0192】次の S 281 では、制御部 28 は、 k が $IRnum$ と等しいか否かをチェックする。そして、もし等しくなければ処理を S 276 に戻す。以上説明した S 276 乃至 S 281 のループを繰り返した結果、 k が $IRnum$ と等しくなった場合には、処理は S 281 からこのループを抜けて S 282 に進む。

【0193】次の S 282 では、制御部 28 は、 n が $Rnum$ と等しいか否かをチェックする。そして、もし等しくなければ処理を S 247 に戻す。以上説明した S 247 乃至 S 282 のループを繰り返した結果、 n が $Rnum$ と等しくなった場合には、処理は S 282 からこのループを抜けて S 283 に進む。

【0194】次の S 283 では、制御部 28 は、 f が $Fnum$ と等しいか否かをチェックする。そして、もし等しくなければ処理を S 208 に戻す。以上説明した S 208 乃至 S 283 のループを繰り返した結果、 f が $Fnum$ と等しくなった場合には、処理は S 283 からこのループを抜けて S 284 に進む。

【0195】次の S 284 では、制御部 28 は、 s が $Snum$ と等しいか否かをチェックする。そして、もし等しくなければ処理を S 206 に戻す。以上説明した S 206 乃至 S 284 のループを繰り返した結果、 s が $Snum$ と等しくなった場合には、処理は S 284 からこのループを抜けて処理を終了する。

【0196】以上のようにして、制御部 28 は、MPEG データ中から、動体領域を抽出するとともに、各動体領域の移動方向ベクトルを算出することができる。また、各動体領域中の色領域を抽出するとともに、各色領域の色に最も近い代表色番号、各色領域の画素数、及び各色領域の重心座標を検出することができる。

<特徴量データの入力処理及び検索処理>本第 2 実施形態の検索データの入力処理及び当該検索データに基づく検索処理は、先述した第 1 実施形態と同様であるので、その説明を省略する。

【0197】

【第 3 実施形態】次に、本発明の第 3 の実施形態について説明する。第 3 の実施形態の回路構成は、先に説明した第 1 の実施形態と同様であり、図 1 によって示される。

10 【0198】次に、第 3 の実施形態の動作について、図 28 乃至 図 43 のフローチャートによって説明する。

<MPEG データの取り込み及び特徴量データ抽出処理>動画像検索システム 10 の制御部 28 は、オペレータによってキーボード 12 から MPEG データ取り込み指示データが入力されると、図 28 乃至 図 43 の処理を実行する。

【0199】図 28 乃至 図 43 の処理を開始すると、最初の S 401 では、動画像検索システム 10 の制御部 28 は、画像インターフェース 24 を介して MPEG データ

20 を受け取り、記憶装置 25 に書き込む。

【0200】次の S 402 では、制御部 28 は、S 401 で受け取った MPEG データ中の全てのカット位置を検出し、記憶装置 25 中のカット情報データベースに書き込む。

【0201】次の S 403 では、制御部 28 は、S 402 で検出したカット位置のデータからショット数を求め、変数 $Snum$ に代入する。次の S 404 では、制御部 28 は、各フレーム中のマクロブロックの数を変数 $Mnum$ に、ブロックの数を変数 $Bnum$ に夫々代入する。

30 【0202】次の S 405 では、制御部 28 は、変数 s, f を 0 に初期化する。次に処理は、S 406 乃至 S 517 のループに入る。このループに入って最初の S 406 では、制御部 28 は、 s を 1 つインクリメントする。

【0203】次の S 407 では、制御部 28 は、第 s ショット中のフレーム数を求め、変数 $Fnum$ に代入する。次の S 408 では、制御部 28 は、配列型変数 $pvx [1..MBnum], pvy [1..MBnum]$ 40 を 0 に初期化する。この pvx は、各マクロブロックの予測動ベクトルの x 成分を保持する変数である。同様に、 pvy は、各マクロブロックの予測動ベクトルの y 成分を保持する変数である。

【0204】次に処理は、S 409 乃至 S 516 のループに入る。このループに入って最初の S 409 では、制御部 28 は、 f を 1 つインクリメントする。

【0205】次の S 410 では、制御部 28 は、記憶装置 25 から第 s ショットの第 f フレーム中の各マクロブロックの動ベクトルを読み出し、その x 成分を配列型変数 $vx [1..MBnum]$ に、 y 成分を $vy [1..$

50 ..]

MBnum]に夫々代入する。

【0206】次のS411では、制御部28は、第fフレームがI(Intra)フレームであるか否かをチェックする。そして、もしIフレームであれば処理をS412へ移し、そうでなければ処理をS431へ移す。

【0207】S412では、制御部28は、動ベクトルに代えて予測動ベクトルを使用するべく、配列型変数pvxの各要素の値をvxの各要素に、pvyの各要素の値をvyの各要素に夫々代入する。

【0208】S413では、制御部28は、第fフレームが第sショットにおける最初のIフレームであるか否かをチェックする。そして、もし最初のIフレームであれば処理をS414へ移し、そうでなければ処理をS416へ移す。

【0209】S414では、制御部28は、第fフレーム中の各マクロブロックのDC成分データを読み出す。尚、各マクロブロック中には、Y(輝度信号)成分のためのブロックが4個ある。従って、Y成分のDC成分も4個があるので、4個の算術平均をY成分のDC成分とする。Cb及びCr(色相信号)成分のためのブロックは夫々1個であるので、夫々のDC成分をそのまま読み出す。

【0210】次のS415では、制御部28は、S414で読み出したDC成分データをL*a*b*表色系に変換し、dcl[1..MBnum], dca[1..MBnum], 及びdcb[1..MBnum]に夫々代入する。

【0211】次のS416では、制御部28は、配列型変数rk[1..MBnum]を0に初期化する。次のS417では、制御部28は、変数mを0に初期化する。

【0212】次に処理は、S418乃至S423のループに入る。このループに入つて最初のS418では、制御部28は、mを1つインクリメントする。

【0213】次のS419では、制御部28は、第fフレーム中の第mマクロブロックのDC成分を読み出してL*a*b*表色系に変換し、その値をml, ma, 及びmbに夫々代入する。

【0214】次のS420では、制御部28は、ml, ma, 及びmbで定まる色とdcl[m], dca[m], 及びdcb[m]で定まる色との色差を算出する。次のS421では、制御部28は、S422で算出した色差が所定の閾値以上であるか否かをチェックする。そして、もし閾値以上であれば処理をS219へ移し、そうでなければ処理をS423へ移す。

【0215】S422では、制御部28は、rk[m]に1を代入する。S423では、制御部28は、mがMBnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS418へ戻す。

【0216】以上説明したS418乃至S423のルー

プを繰り返した結果、mがMBnumと等しくなった場合には処理はS423からこのループを抜けて、S424に進む。

【0217】次のS424では、制御部28は、変数mを0に初期化する。次に処理は、S425乃至S430のループに入る。このループに入つて最初のS425では、制御部28は、mを1つインクリメントする。

【0218】次のS426では、制御部28は、rk[m]が1であるか否かをチェックする。そして、もし1であれば処理をS430へ移し、そうでなければ処理をS427へ移す。

【0219】S427では、制御部28は、第mマクロブロックを囲む8個のマクロブロックを求める。次のS428では、制御部28は、配列型変数rkの要素のうちS427で求めた8個のマクロブロックに対応するものの値が全て0であるか否かをチェックする。そして、もし全て0であれば処理をS429へ移し、そうでなければ処理をS430へ移す。

【0220】S429では、制御部28は、{dcl[m]* (N-1) + ml} / Nを算出してdcl[m]に代入する。同様に、{dca[m]* (N-1) + ma} / Nを算出してdca[m]に代入し、{dcb[m]* (N-1) + mb} / Nを算出してdcb[m]に代入する。尚、第sショットの処理を開始してから現れたIフレームが未だN個に満たない場合には、上式のNをIフレームが現れた数によって置き換えて算出する。

【0221】次のS430では、制御部28は、mがMBnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS425へ戻す。以上説明したS425乃至S430のループを繰り返した結果、mがMBnumと等しくなった場合には処理はS430からこのループを抜けて、S460に進む。

【0222】これに対し、S411にてIフレームでないと判断した場合、次のS431において、制御部28は、配列型変数rk[1..MBnum]及び変数mを0に初期化する。

【0223】次に処理は、S432乃至S437のループに入る。このループに入つて最初のS432では、制御部28は、mを1つインクリメントする。

【0224】次のS433では、制御部28は、第sショットの第fフレーム中の第mマクロブロックがIntra-codedマクロブロックであるか否かをチェックする。そして、もしIntra-codedマクロブロックであれば処理をS436へ移し、そうでなければ処理をS434へ移す。

【0225】S434では、制御部28は、vx[m]及びvy[m]のデータからベクトル長を算出する。次のS435では、制御部28は、S435で算出したベクトル長が所定の閾値以上であるか否かをチェックす

る。そして、もし閾値以上であれば処理をS436へ移し、そうでなければ処理をS437へ移す。

【0226】S433にて第mマクロブロックがInt r a - c o d e d マクロブロックであると判断した場合又はS435にてベクトル長が閾値以上であると判断した場合には、次のS436において、制御部28は、rk[m]に1を代入する。その後、処理はS437へ進む。

【0227】S437では、制御部28は、mの値がMBnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS432へ戻す。以上説明したS432乃至S437のループを繰り返した結果、mがMBnumと等しくなった場合には処理はS437からこのループを抜けて、S438に進む。

【0228】次のS438では、制御部28は、変数mを0に初期化する。次に処理は、S439乃至S444のループに入る。このループに入って最初のS439では、制御部28は、mを1つインクリメントする。

【0229】次のS440では、制御部28は、rk[m]が1であるか否かをチェックする。そして、もし1であれば処理をS441へ移し、そうでなければ処理をS444へ移す。

【0230】S441では、制御部28は、第mマクロブロックを囲む8個のマクロブロックを求める。次のS442では、制御部28は、配列型変数rkの要素のうちS441で求めた8個のマクロブロックに対応する要素の値が全て0であるか否かを、チェックする。そして、もし全て0であれば処理をS443へ移し、そうでなければ処理をS444へ移す。

【0231】S443では、制御部28は、rk[m]に0を代入する。その後、処理はS444へ進む。S444では、制御部28は、mの値がMBnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS439へ戻す。

【0232】以上説明したS439乃至S444のループを繰り返した結果、mがMBnumと等しくなった場合には処理はS444からこのループを抜けて、S445に進む。

【0233】次のS445では、制御部28は、変数mを0に初期化する。次に処理は、S446乃至S451のループに入る。このループに入って最初のS446では、制御部28は、mを1つインクリメントする。

【0234】次のS447では、制御部28は、rk[m]が1であるか否かをチェックする。そして、もし1でなければ処理をS448へ移し、そうでなければ処理をS451へ移す。

【0235】S448では、制御部28は、配列型変数rkの要素であって画面上第mマクロブロックの左上、上、及び右上に位置する3個のマクロブロックに対応するもののうち2つ以上が1であり、且つ、rkの要素で

あって画面上第mマクロブロックの左下、下、及び右下に位置する3個のマクロブロックに対応するもののうち2つ以上が1であるか否かをチェックする。そして、もし以上の条件を満たさない場合には処理をS449へ移し、満たす場合には処理をS450へ移す。

【0236】S449では、制御部28は、配列型変数rkの要素であって画面上第mマクロブロックの左上、左、及び左下に位置する3個のマクロブロックに対応するもののうち2つ以上が1であり、且つ、rkの要素であって画面上第mマクロブロックの右上、右、及び右下に位置する3個のマクロブロックに対応するもののうち2つ以上が1であるか否かをチェックする。そして、もし以上の条件を満たす場合には処理をS450へ移し、満たさない場合には処理をS451へ移す。

【0237】S450では、制御部28は、rk[m]に1を代入する。その後、処理はS451へ進む。次のS451では、制御部28は、mの値がMBnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS446へ戻す。

【0238】以上説明したS446乃至S451のループを繰り返した結果、mがMBnumと等しくなった場合には処理はS451からこのループを抜けて、S452に進む。

【0239】次のS452では、制御部28は、変数mを0に初期化する。次に処理は、S453乃至S459のループに入る。このループに入って最初のS453では、制御部28は、mを1つインクリメントする。

【0240】次のS454では、制御部28は、第fフレーム中の第mマクロブロックの画面上のx座標を変数mxに、y座標を変数myに夫々代入する。次のS455では、制御部28は、マクロブロックの幅を変数mwに代入する。

【0241】次のS456では、制御部28は、x座標がmx+vx[m]/mw、y座標がmy+vy[m]/mwであるマクロブロックを第fフレーム中から抽出する。

【0242】次のS457では、制御部28は、S456で抽出したマクロブロックの番号を変数nに代入する。次のS458では、制御部28は、vx[n]+vx[m]/DAMPを計算して予測動ベクトルのx成分とし、pxx[n]に代入する。同様に、vy[n]+vy[m]/DAMPを計算して予測動ベクトルのy成分とし、pyy[n]に代入する。ここでDAMPは、所定の定数である。

【0243】次のS459では、制御部28は、mの値がMBnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS453へ戻す。以上説明したS453乃至S459のループを繰り返した結果、mがMBnumと等しくなった場合には処理はS459からこのループを抜けて、S460に進む。

【0244】次のS460では、制御部28は、配列型変数r_kの要素の値が1であるものに対応するマクロブロックが連接する領域を動体領域候補として検出し、夫々に仮のラベル番号を付与する。

【0245】次のS461では、制御部28は、S460で検出した動体領域候補の数を変数DRKn umに代入する。次のS462では、制御部28は、nを0に初期化する。

【0246】次に処理は、S463乃至S475のループに入る。このループに入って最初のS463では、制御部28は、nを1つインクリメントする。

【0247】次のS464では、制御部28は、第n動体領域候補中のマクロブロックの数を変数MBDRKn umに代入する。次のS465では、制御部28は、MBDRKn umの値が所定の閾値以下であるか否かをチェックする。そして、閾値以下であれば処理をS466に移し、そうでなければ処理をS467に移す。

【0248】S466では、制御部28は、第n動体領域候補を構成する各マクロブロックに対するr_kの要素に0を代入する。そして、処理はその後S475に進む。これに対し、S465にてMBDRKn umの値が所定の閾値以下でないと判断した場合には、次のS467において、制御部28は、第fフレーム中において第n動体領域候補が占める範囲に相当する第(f-1)フレーム中の範囲に、動体領域があるか否かをチェックする。そして、もしそのような動体領域があれば処理をS469へ移し、動体領域がなければ処理をS468へ移す。

【0249】S468では、制御部28は、第n動体領域候補に新たなラベル番号を付与する。そして、そのラベル番号を確定したラベル番号とする。その後、処理はS470へ進む。

【0250】これに対し、S467にて動体領域があると判断した場合には、次のS469において、制御部28は、S467で見つけ出した動体領域のうち第n動体領域候補が占める範囲に相当する範囲に最も重なっている動体領域を選択する。そして、当該動体領域に付与されたラベル番号を第n動体領域候補に付与し、確定したラベル番号とする。その後、処理はS470へ進む。

【0251】次のS470では、制御部28は、変数ax, ay, c, mを0に初期化する。次に処理は、S471乃至S473のループに入る。

【0252】このループに入って最初のS471では、制御部28は、mを1つインクリメントする。次のS472では、制御部28は、axにvx[m]を加算し、ayにvy[m]を加算する。更に、cをインクリメントする。

【0253】次のS473では、制御部28は、mがMBDRKn umと等しいか否かをチェックする。そして、もし等しくなければ処理をS471へ戻す。以上説

明したS471乃至S473のループを繰り返した結果、mがMBDRKn umと等しくなった場合には処理はS473からこのループを抜けて、S474に進む。

【0254】次のS474では、制御部28は、第n動体領域候補の移動方向ベクトルのx成分をax/c、y成分をay/cとして算出する。そして、このように算出されたax/c及びay/cの値を、S468又はS469で第n動体領域候補に付与されたラベル番号に対応させて、記憶装置25中の特徴量データベースに書き込む。

【0255】次のS475では、制御部28は、nがDRKn umと等しいか否かをチェックする。そして、もし等しくなければ処理をS463へ戻す。以上説明したS463乃至S475のループを繰り返した結果、nがDRKn umと等しくなった場合には、処理はS475からこのループを抜けてS476に進む。

【0256】次のS476では、制御部28は、第fフレームのデータを画像データに復号化する。次のS477では、制御部28は、S476で得た画像データをL*a*b*表色系に基づく画像データに変換する。

【0257】次のS478では、制御部28は、配列型変数r_kの要素のうち値が1であるものに対応するマクロブロックが連接する動体領域を検出し、その数を変数Rnumに代入する。

【0258】次のS479では、制御部28は、変数nを0に初期化する。次に処理は、S480乃至S515のループに入る。このループに入って最初のS480では、制御部28は、nを1つインクリメントする。

【0259】次のS481では、制御部28は、所定数(C個)のL*a*b*表色系に基づく代表色の初期値を乱数によって夫々生成する。そして、これらの代表色のL*を配列型変数I[1..C]に、a*をa[1..C]に、b*をb[1..C]に夫々代入する。

【0260】次のS482では、制御部28は、変数iを0に初期化する。次に、処理は、S483乃至S501のループに入る。このループに入って最初のS483では、制御部28は、iをインクリメントする。

【0261】次のS484では、制御部28は、第n動体領域内の画素の数を変数Gnumに代入する。次のS485では、制御部28は、変数gを0に初期化する。

【0262】次に、処理は、S486乃至S488のループに入る。このループ进入到最初のS486では、制御部28は、gをインクリメントする。

【0263】次のS487では、制御部28は、第g画素の色に最も近い色をC個の代表色の中から選択し、その番号を配列型変数ds[g]に代入する。尚、色の近さは、L*, a*, 及びb*の値によって張られる3次元空間におけるユークリッド距離の大小によって判断する。以下において色の近さを判断するときも、同様とす

る。

【0264】次のS488では、制御部28は、gがGnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS486に戻す。以上説明したS486乃至S488のループを繰り返した結果、gがGnumと等しくなった場合には、処理はS488からこのループを抜けてS489に進む。

【0265】次のS489では、制御部28は、変数cを0に初期化する。次に、処理は、S490乃至S498のループに入る。このループに入って最初のS490では、制御部28は、cをインクリメントする。

【0266】次のS491では、制御部28は、変数g, c1, ca, cb, chを0に初期化する。次に、処理は、S492乃至S496のループに入る。

【0267】このループに入って最初のS492では、制御部28は、gをインクリメントする。次のS493では、制御部28は、ds[g]がcと等しいか否かをチェックする。そして、もし等しければ処理をS494へ移し、そうでなければ処理をS496へ移す。

【0268】次のS494では、制御部28は、第g画素のa*を変数gaに、b*をgbに、L*をglに夫々代入する。次のS495では、制御部28は、c1にglを加算し、caにgaを加算し、cbにgbを加算する。更に、chをインクリメントする。

【0269】次のS496では、制御部28は、gがGnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS492に戻す。以上説明したS492乃至S496のループを繰り返した結果、gがGnumと等しくなった場合には、処理はS496からこのループを抜けてS497に進む。

【0270】次のS497では、制御部28は、c1/chを算出して1[c]に代入し、ca/chを算出してa[c]に代入し、cb/chを算出してb[c]に代入する。更に、h[c]にchを代入する。

【0271】次のS498では、制御部28は、cがCと等しいか否かをチェックする。そして、もし等しくなければ処理をS490に戻す。以上説明したS490乃至S498のループを繰り返した結果、cがCと等しくなった場合には、処理はS498からこのループを抜けてS499に進む。

【0272】次のS499では、制御部28は、iが所定の繰り返し回数の値に達したか否かをチェックする。そして、もし達していないければ処理をS500に移し、達していればS483乃至S501のループを抜けて処理をS502へ移す。

【0273】S500では、制御部28は、配列型変数hの要素の最大値の添字を変数maxに代入する。また、最小値の添字を変数minに代入する。次のS501では、制御部28は、l[max]の値に乱数を加算した値をl[min]に代入する。同様に、a[min]

x]の値に乱数を加算した値をa[min]に、b[min]の値に乱数を加算した値をb[min]に夫々代入する。尚、乱数の値は、l[max], a[max], 及びb[max]の値に比して小さな値とする。その後、処理はS483へ戻る。

【0274】これに対し、S499にて、iが所定の繰り返し回数の値に達したと判断した場合、次のS502において、制御部28は、変数gを0に初期化する。次に、処理は、S503乃至S505のループに入る。

10 【0275】このループに入って最初のS503では、制御部28は、gをインクリメントする。次のS504では、制御部28は、第g画素の色に最も近い色をC個の代表色の中から選択し、その代表色の番号を配列型変数ds[g]に代入する。

【0276】次のS505では、制御部28は、gがGnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS503に戻す。以上説明したS503乃至S505のループを繰り返した結果、gがGnumと等しくなった場合には、処理はS505からこのループを抜けてS506に進む。

20 【0277】次のS506では、制御部28は、配列型変数dsの値が等しい要素に対応するマクロブロックが連接する領域を、色領域として検出する。次のS507では、制御部28は、S506で検出した色領域の数を変数IRnumに代入する。

【0278】次のS508では、制御部28は、変数kを0に初期化する。次に、処理は、S509乃至S514のループに入る。このループに入って最初のS509では、制御部28は、kをインクリメントする。

30 【0279】次のS510では、制御部28は、ノイズのカット等のため、第k色領域を縮小する。次のS511では、制御部28は、第k色領域を拡大する。

【0280】次のS512では、制御部28は、第k色領域を構成する画素数が所定の閾値以上であるか否かをチェックする。そして、もし閾値以上であれば処理をS513へ移し、そうでなければ処理をS514へ移す。

【0281】S513では、制御部28は、第k色領域の色に最も近い代表色の番号、第k色領域の画素数、及び第k色領域の重心座標を、第n動体領域のラベル番号40 であるnに対応させて、記憶装置25中の特徴量データベースに書き込む。

【0282】次のS514では、制御部28は、kがIRnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS509に戻す。以上説明したS509乃至S514のループを繰り返した結果、kがIRnumと等しくなった場合には、処理はS514からこのループを抜けてS515に進む。

【0283】次のS515では、制御部28は、nがRnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS480に戻す。以上説明したS450

80乃至S515のループを繰り返した結果、nがRnumと等しくなった場合には、処理はS515からこのループを抜けてS516に進む。

【0284】次のS516では、制御部28は、fがFnumと等しいか否かをチェックする。そして、もし等しくなければ処理をS409に戻す。以上説明したS409乃至S516のループを繰り返した結果、fがFnumと等しくなった場合には、処理はS516からこのループを抜けてS517に進む。

【0285】次のS517では、制御部28は、sがSunumと等しいか否かをチェックする。そして、もし等しくなければ処理をS406に戻す。以上説明したS406乃至S517のループを繰り返した結果、sがSunumと等しくなった場合には、処理はS516からこのループを抜けて処理を終了する。

【0286】以上のようにして、制御部28は、MPEGデータ中から、動体領域を抽出するとともに、各動体領域の移動方向ベクトルを算出することができる。また、各動体領域中の色領域を抽出するとともに、各色領域の色に最も近い代表色番号、各色領域の画素数、及び各色領域の重心座標を検出することができる。

<特徴量データの入力処理及び検索処理>本第3実施形態の検索データの入力処理及び当該検索データに基づく検索処理は、先述した第1実施形態と同様であるので、その説明を省略する。

【0287】

【発明の効果】以上説明したように、本発明によれば、動画像データから動体領域を抽出するとともに、各動体領域の移動方向ベクトルを算出することができる。また、各動体領域中の色領域を抽出するとともに、各色領域の色に最も近い代表色番号、各色領域の画素数、及び各色領域の重心座標を検出することができる。このようにして抽出された特徴量データに基づいて動画像検索が行われるので、動体領域以外の背景の影響を受けることなく動画像検索を行うことができる。

【図面の簡単な説明】

【図1】本発明の実施の形態による動画像検索システムの構成を示すブロック図

【図2】図1の制御部において実行される制御処理を示すフローチャート

【図3】図1の制御部において実行される制御処理を示すフローチャート

【図4】図1の制御部において実行される制御処理を示すフローチャート

【図5】図1の制御部において実行される制御処理を示すフローチャート

【図6】図1の制御部において実行される制御処理を示すフローチャート

【図7】図1の制御部において実行される制御処理を示すフローチャート

【図8】図1の制御部において実行される制御処理を示すフローチャート

【図9】図1の制御部において実行される制御処理を示すフローチャート

【図10】図1の制御部において実行される制御処理を示すフローチャート

【図11】図1の制御部において実行される制御処理を示すフローチャート

【図12】図1の制御部において実行される制御処理を示すフローチャート

【図13】図1の制御部において実行される制御処理を示すフローチャート

【図14】図1の制御部において実行される制御処理を示すフローチャート

【図15】図1の制御部において実行される制御処理を示すフローチャート

【図16】図1の制御部において実行される制御処理を示すフローチャート

【図17】図1の制御部において実行される制御処理を示すフローチャート

【図18】図1の制御部において実行される制御処理を示すフローチャート

【図19】図1の制御部において実行される制御処理を示すフローチャート

【図20】図1の制御部において実行される制御処理を示すフローチャート

【図21】図1の制御部において実行される制御処理を示すフローチャート

【図22】図1の制御部において実行される制御処理を示すフローチャート

【図23】図1の制御部において実行される制御処理を示すフローチャート

【図24】図1の制御部において実行される制御処理を示すフローチャート

【図25】図1の制御部において実行される制御処理を示すフローチャート

【図26】図1の制御部において実行される制御処理を示すフローチャート

【図27】図1の制御部において実行される制御処理を示すフローチャート

【図28】図1の制御部において実行される制御処理を示すフローチャート

【図29】図1の制御部において実行される制御処理を示すフローチャート

【図30】図1の制御部において実行される制御処理を示すフローチャート

【図31】図1の制御部において実行される制御処理を示すフローチャート

【図32】図1の制御部において実行される制御処理を示すフローチャート

50 示すフローチャート

【図33】図1の制御部において実行される制御処理を示すフローチャート

【図34】図1の制御部において実行される制御処理を示すフローチャート

【図35】図1の制御部において実行される制御処理を示すフローチャート

【図36】図1の制御部において実行される制御処理を示すフローチャート

【図37】図1の制御部において実行される制御処理を示すフローチャート

【図38】図1の制御部において実行される制御処理を示すフローチャート

【図39】図1の制御部において実行される制御処理を示すフローチャート

【図40】図1の制御部において実行される制御処理を示すフローチャート

【図41】図1の制御部において実行される制御処理を示すフローチャート

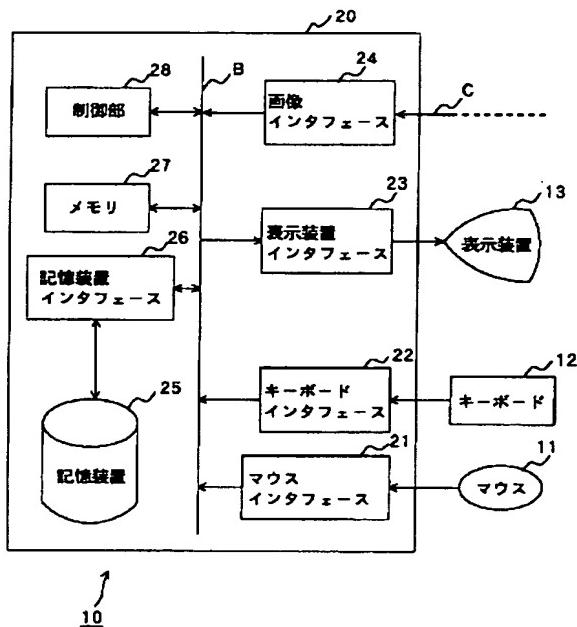
【図42】図1の制御部において実行される制御処理を示すフローチャート

【図43】図1の制御部において実行される制御処理を示すフローチャート

【符号の説明】

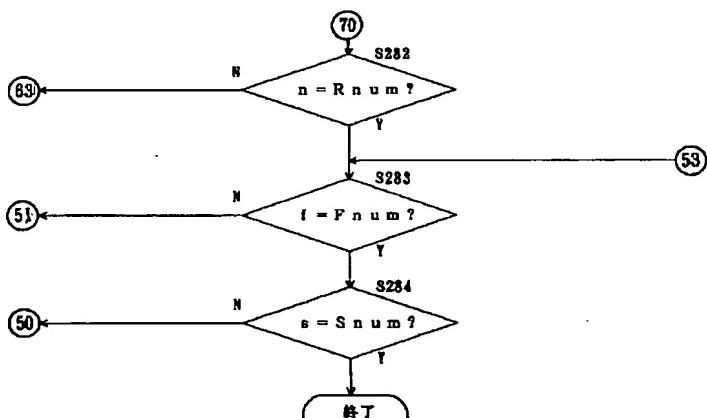
- | | |
|----|---------------|
| 10 | 動画像検索システム |
| 11 | マウス |
| 12 | キーボード |
| 13 | 表示装置 |
| 10 | コンピュータ本体 |
| 21 | マウスインターフェース |
| 22 | キーボードインターフェース |
| 23 | 表示装置インターフェース |
| 24 | 画像インターフェース |
| 25 | 記憶装置インターフェース |
| 26 | メモリ |
| 27 | 記憶装置 |
| 28 | 制御部 |
| 29 | 終了 |

【図1】

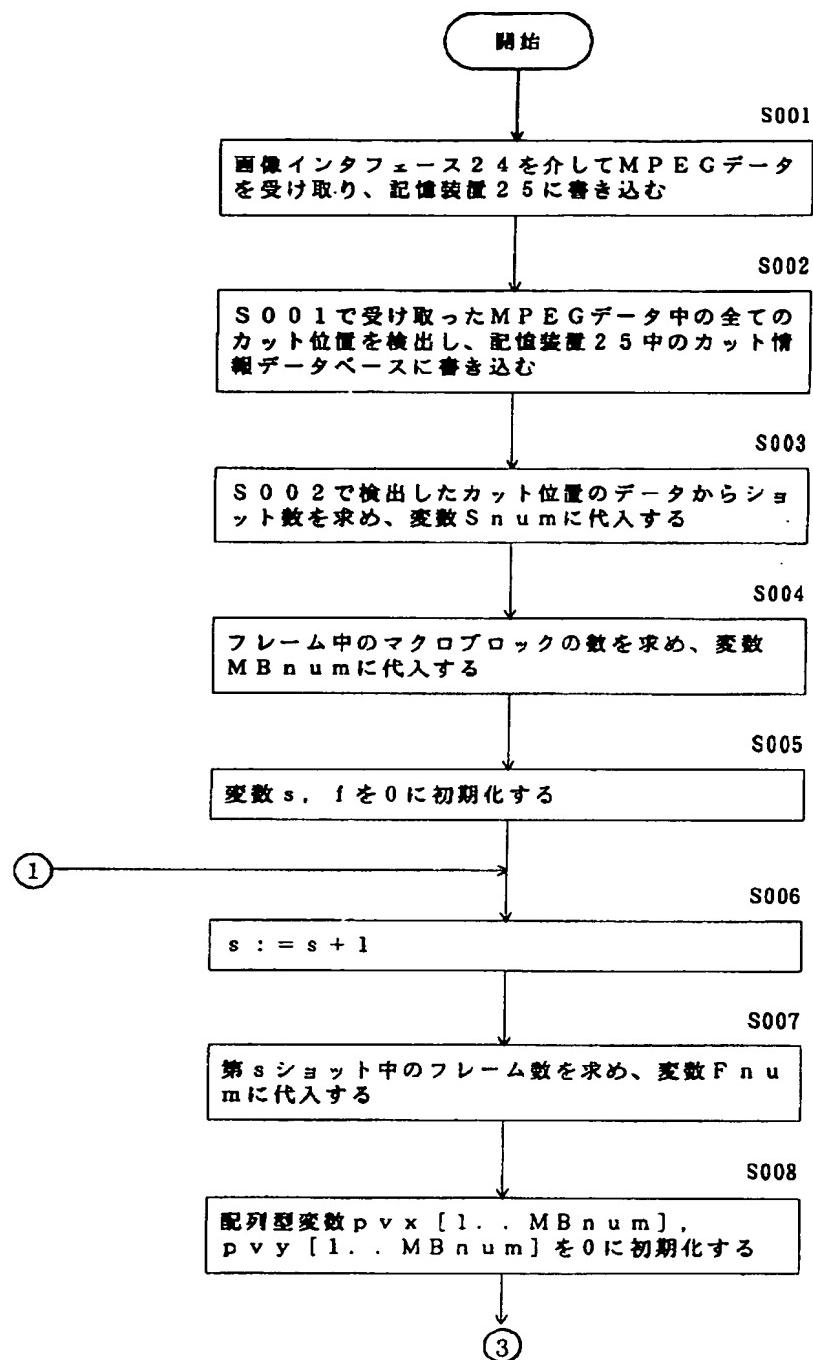


10

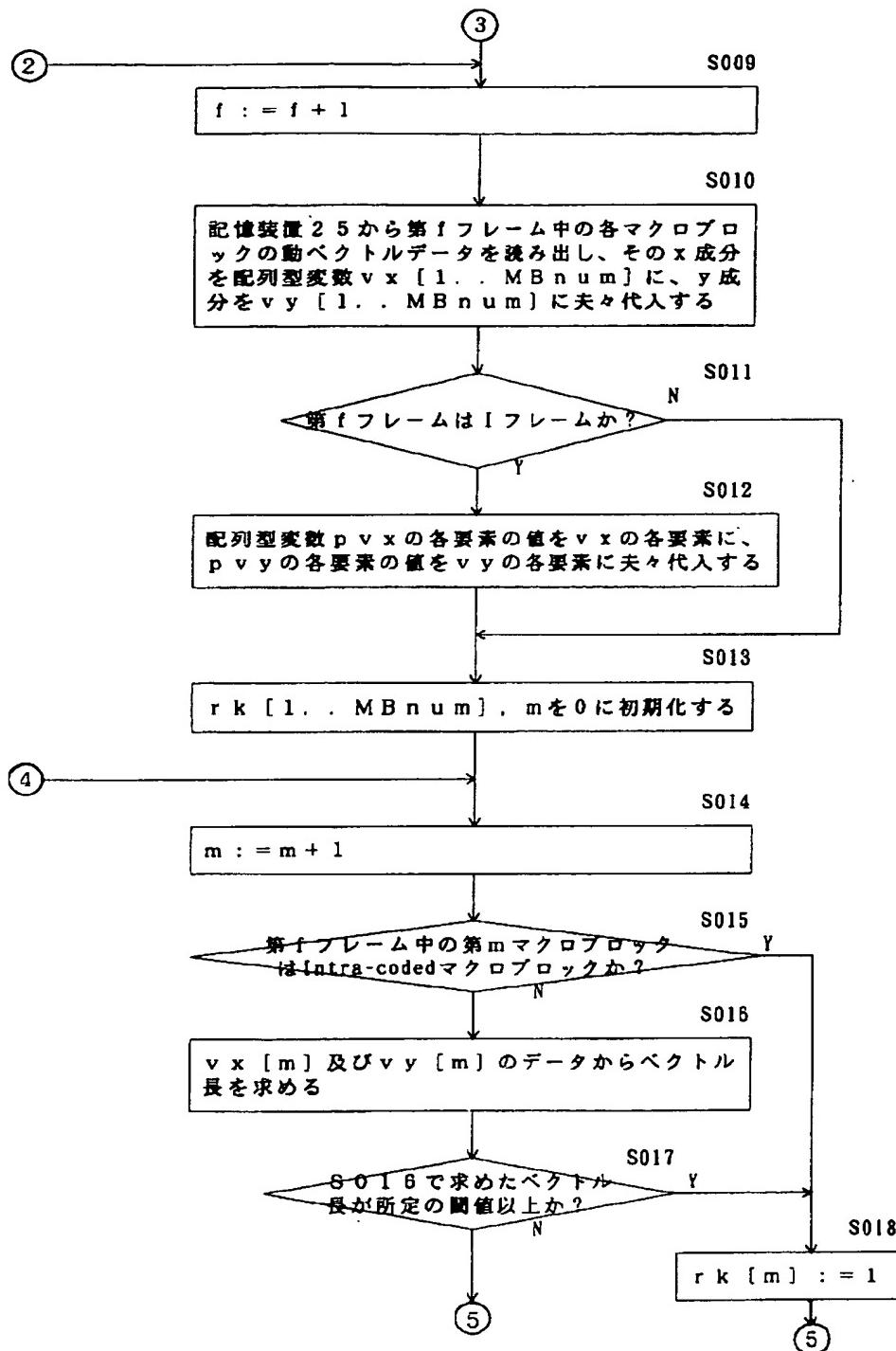
【図27】



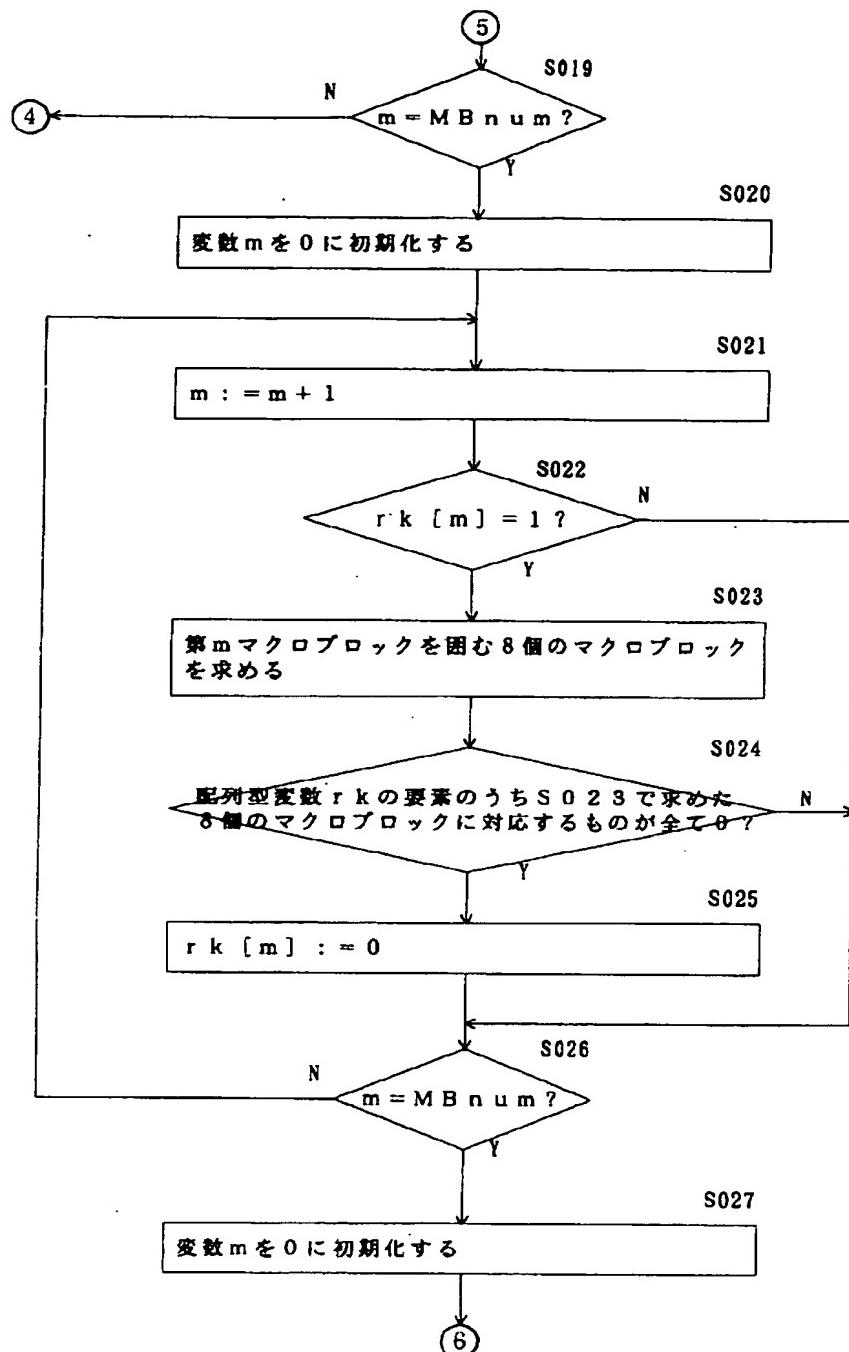
【図2】



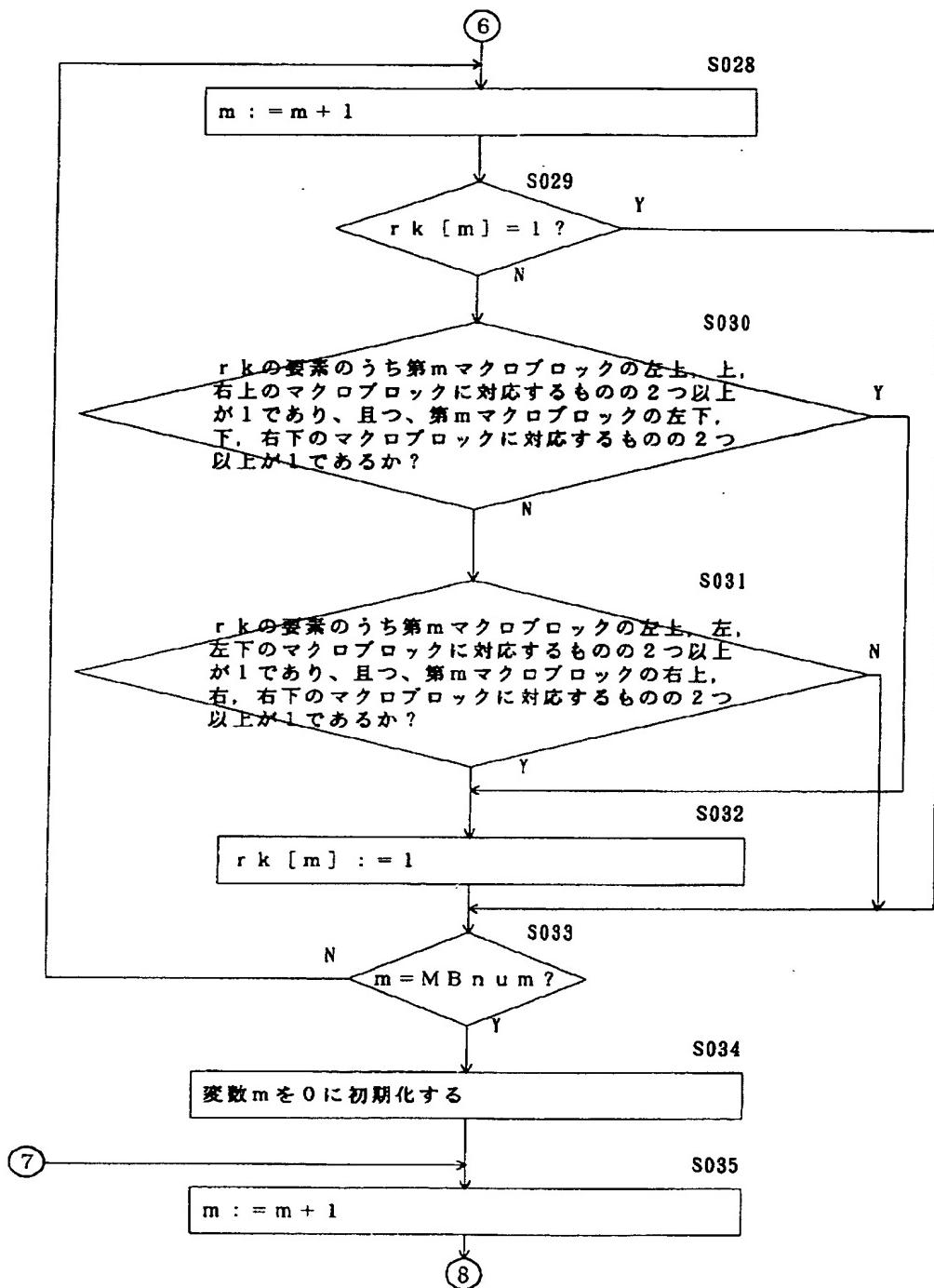
【図3】



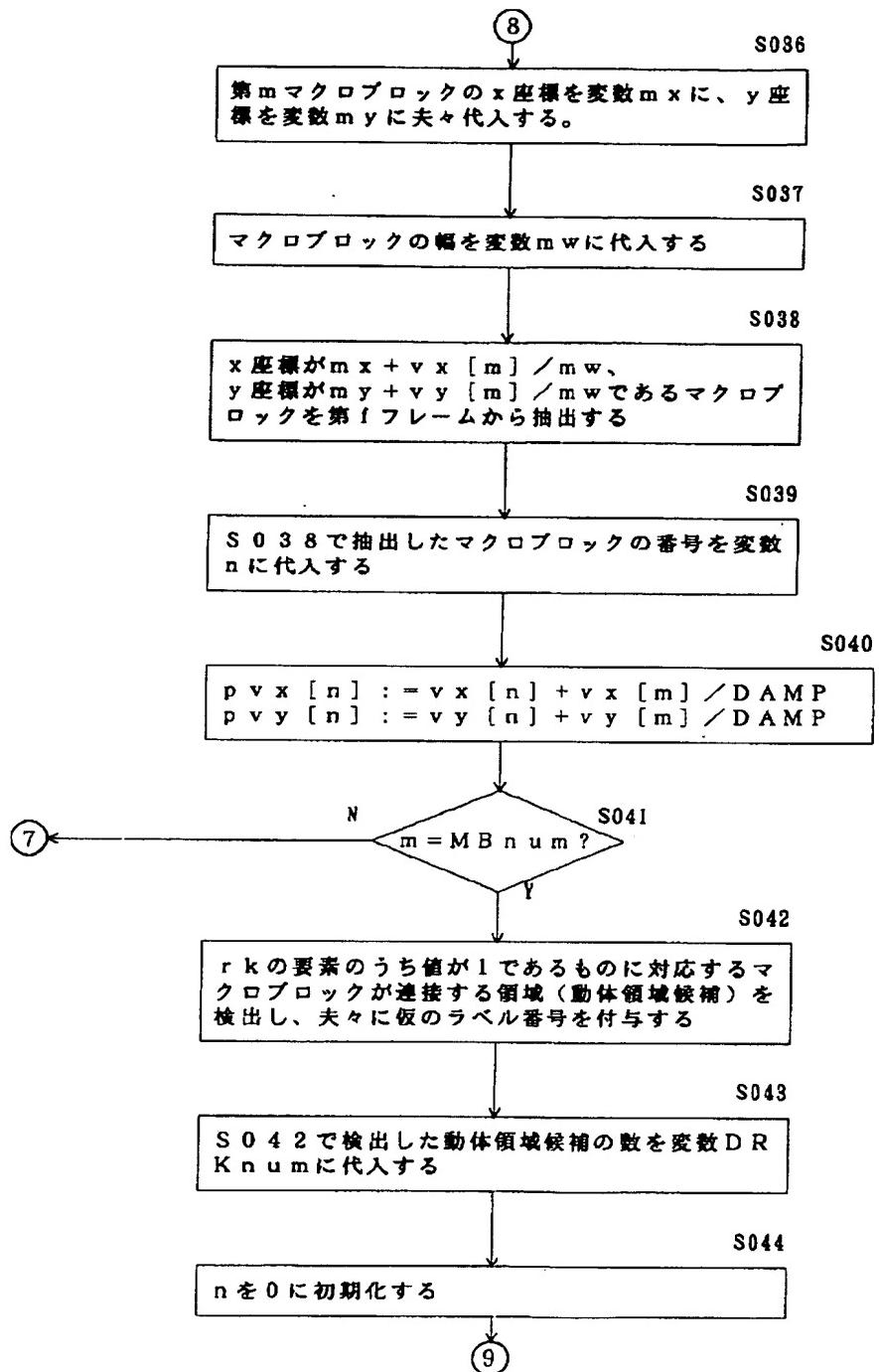
【図4】



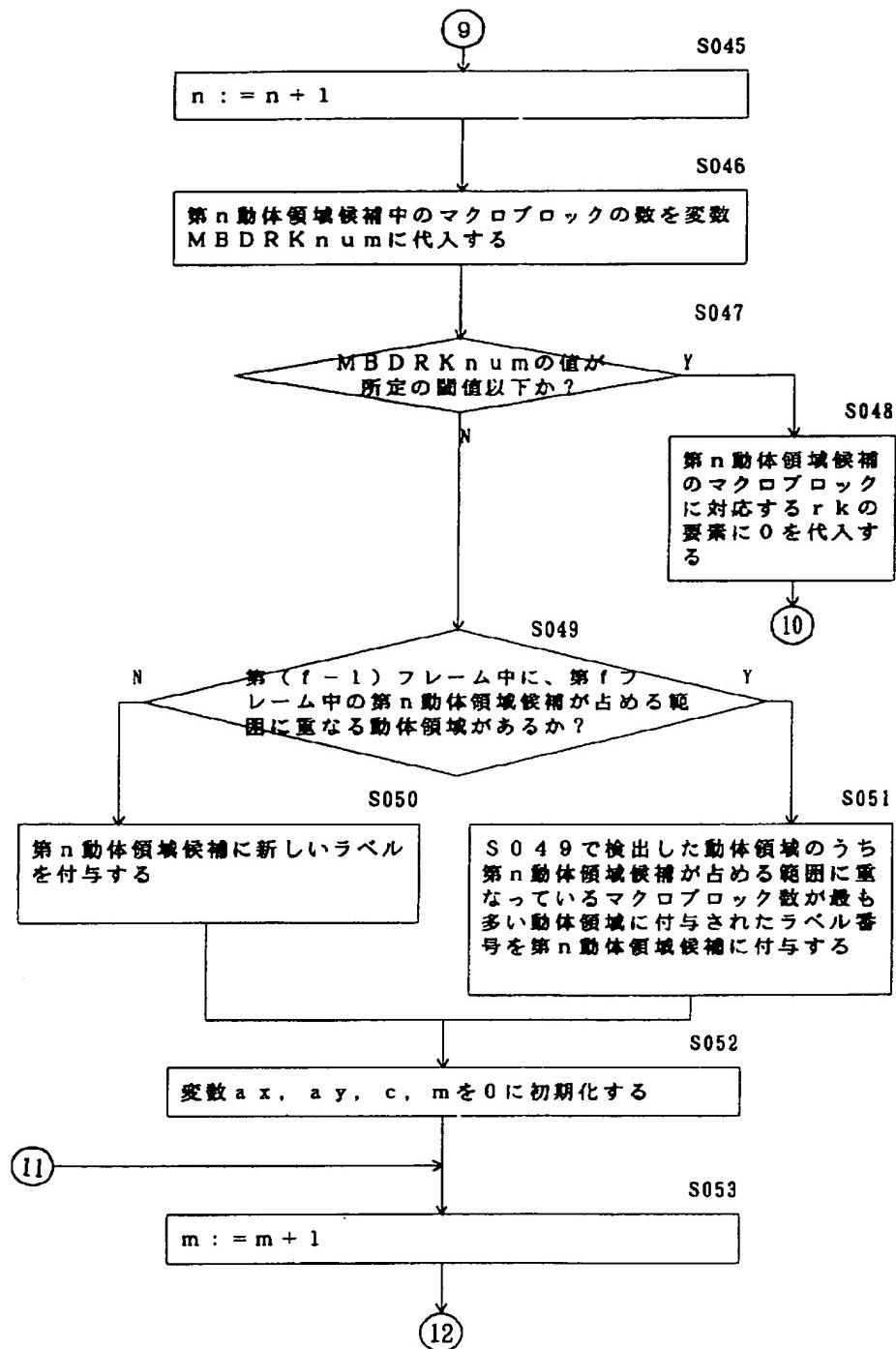
【図5】



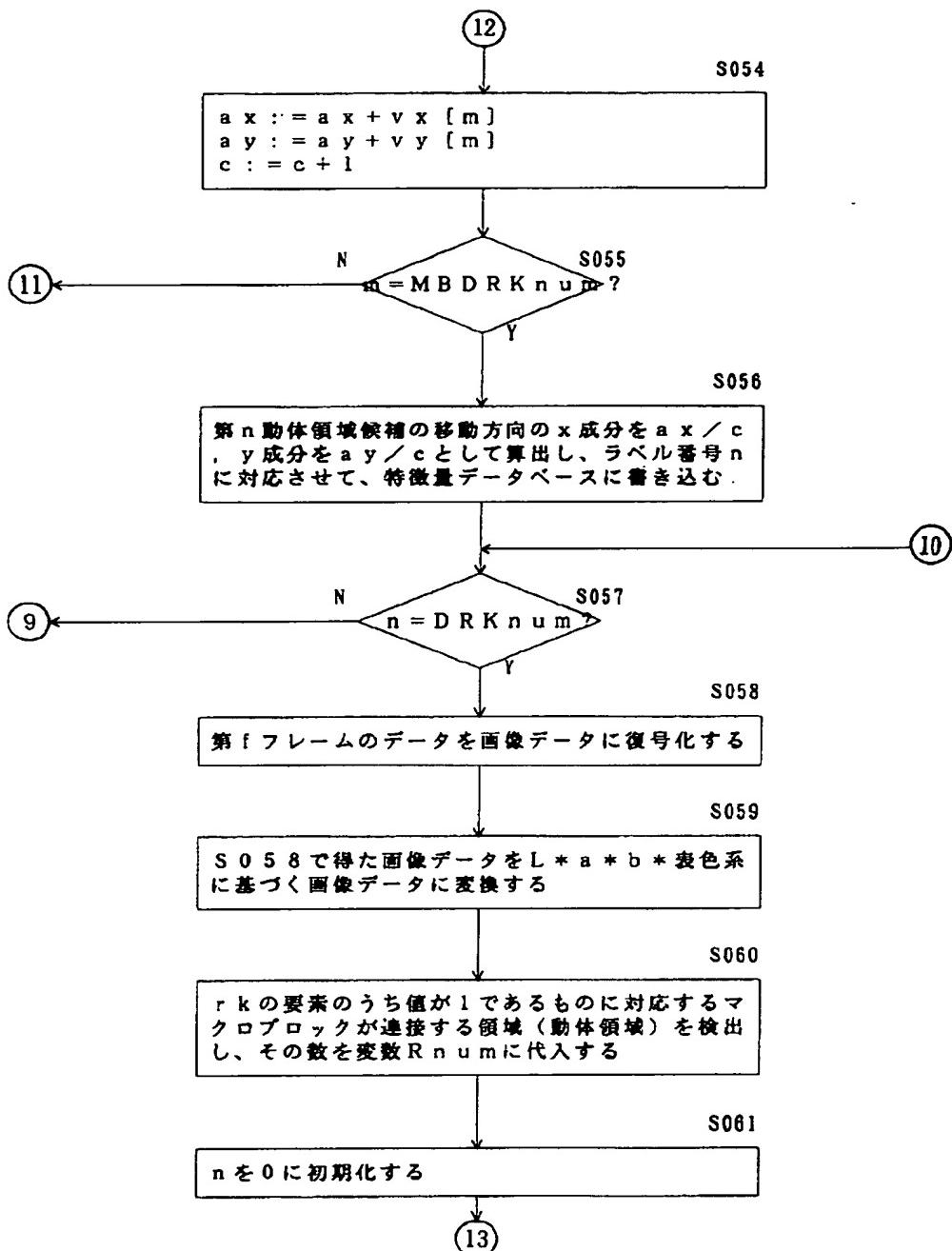
[図6]



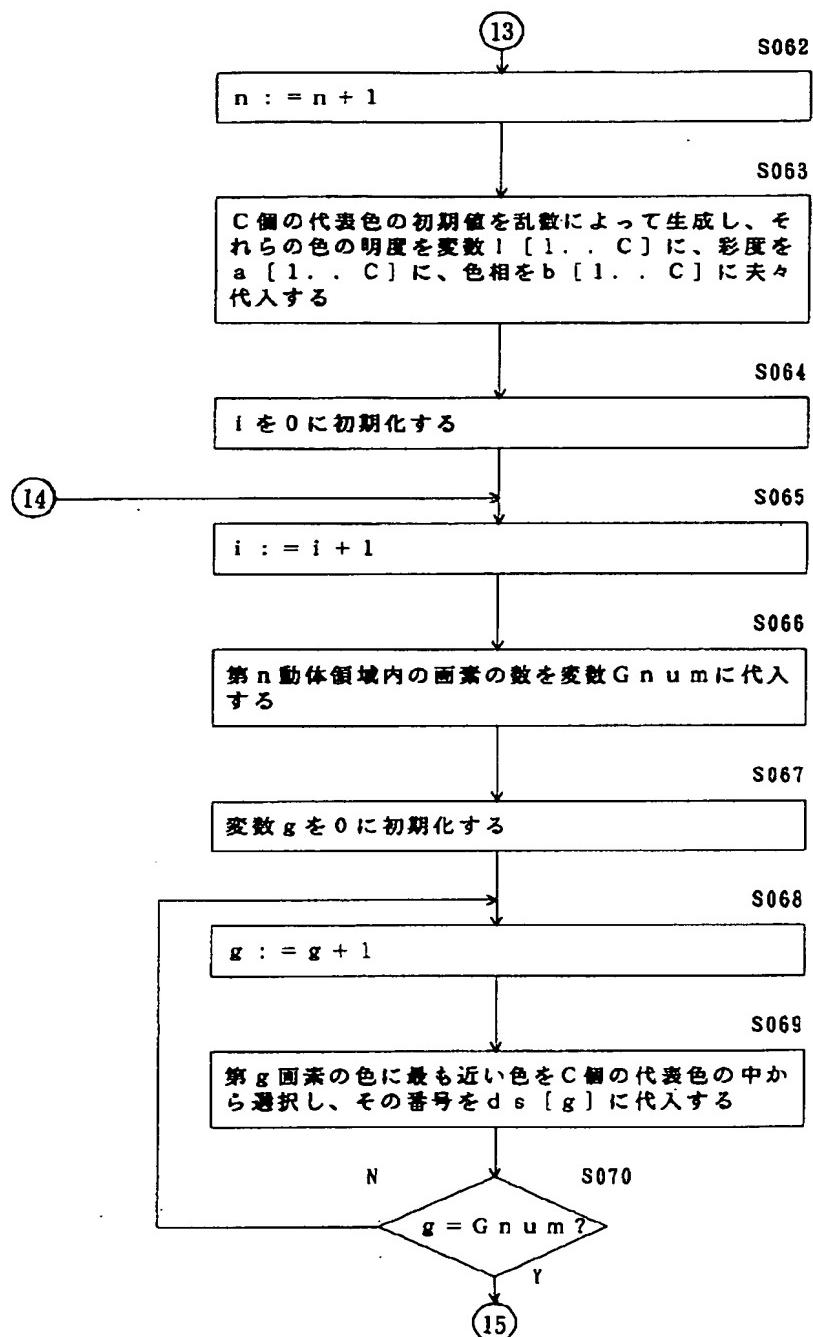
【図7】



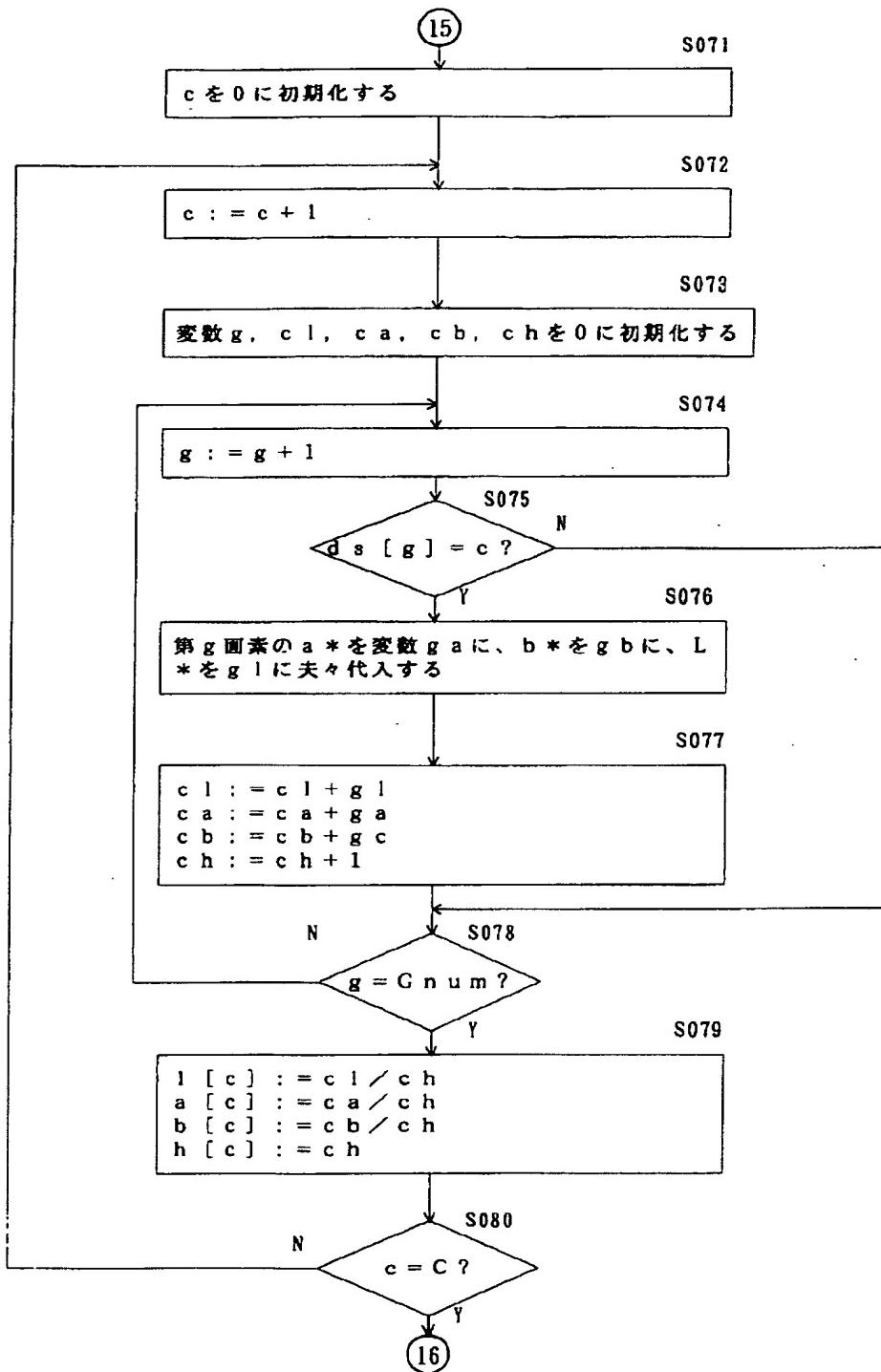
【図8】



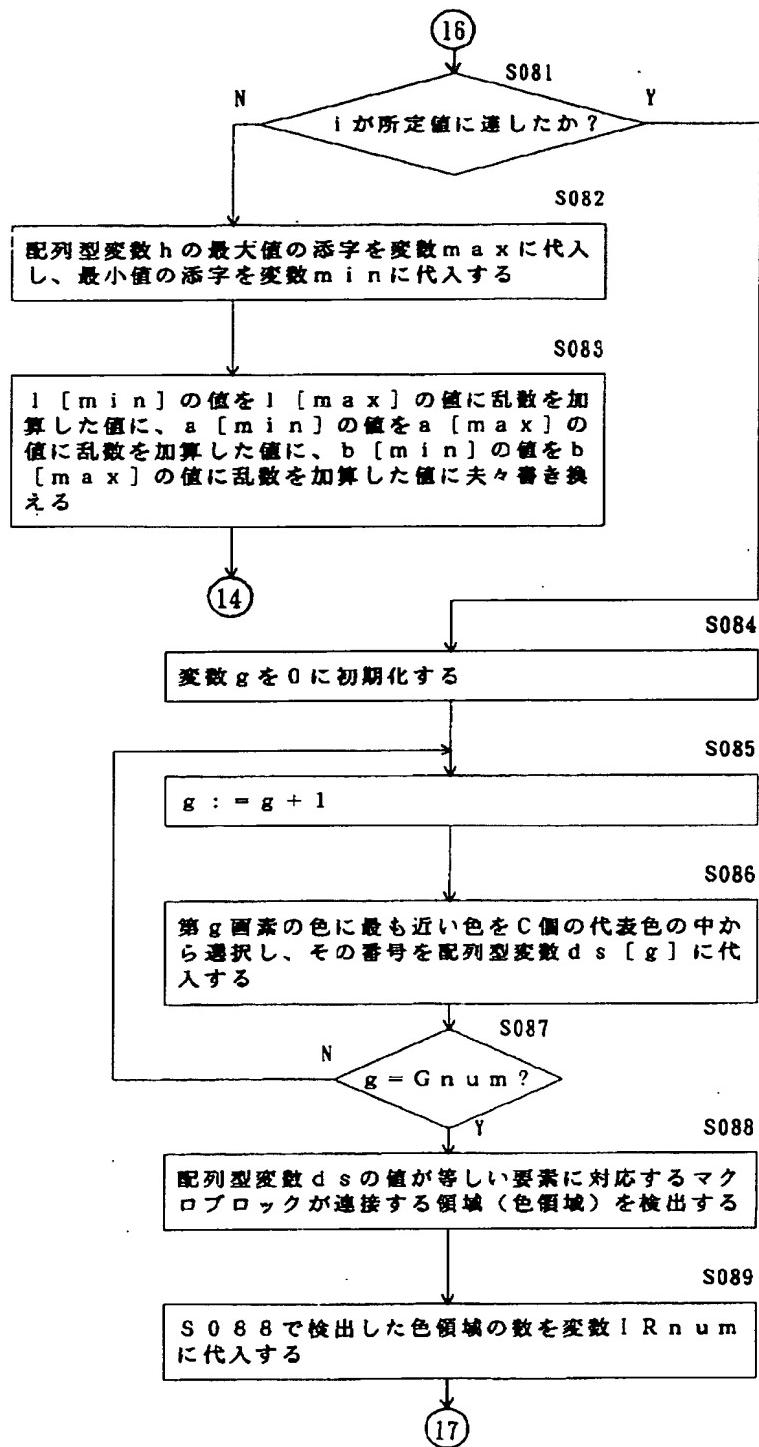
【図9】



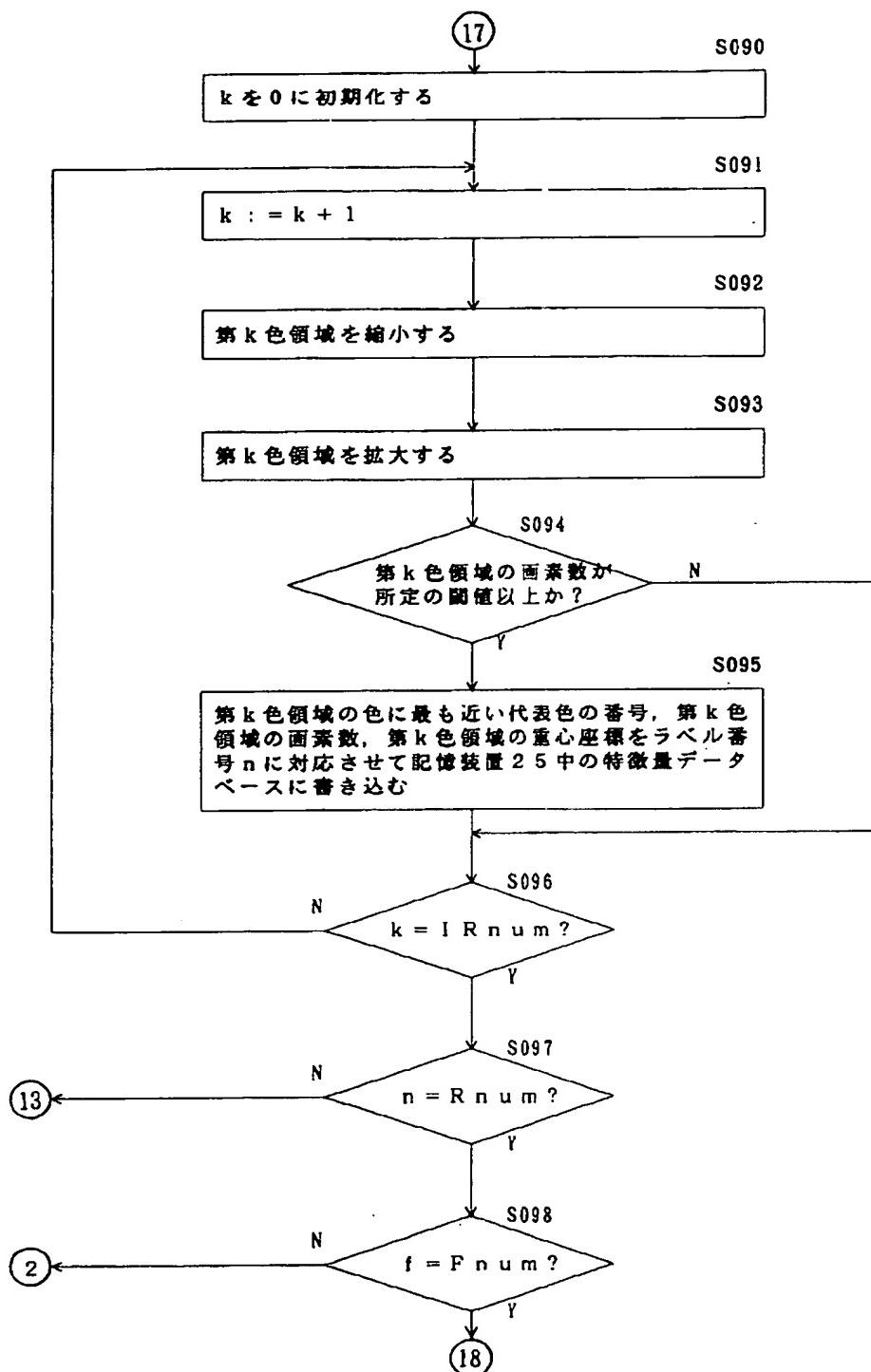
【図10】



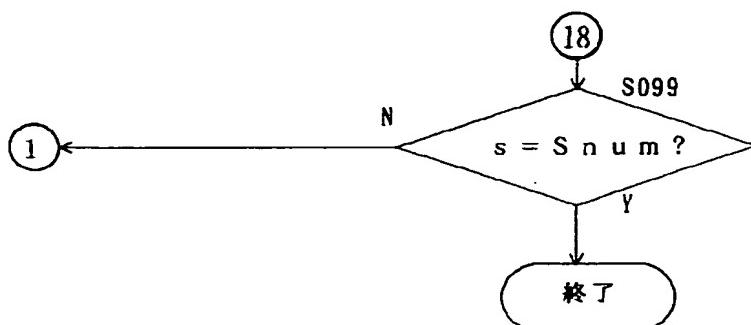
【図11】



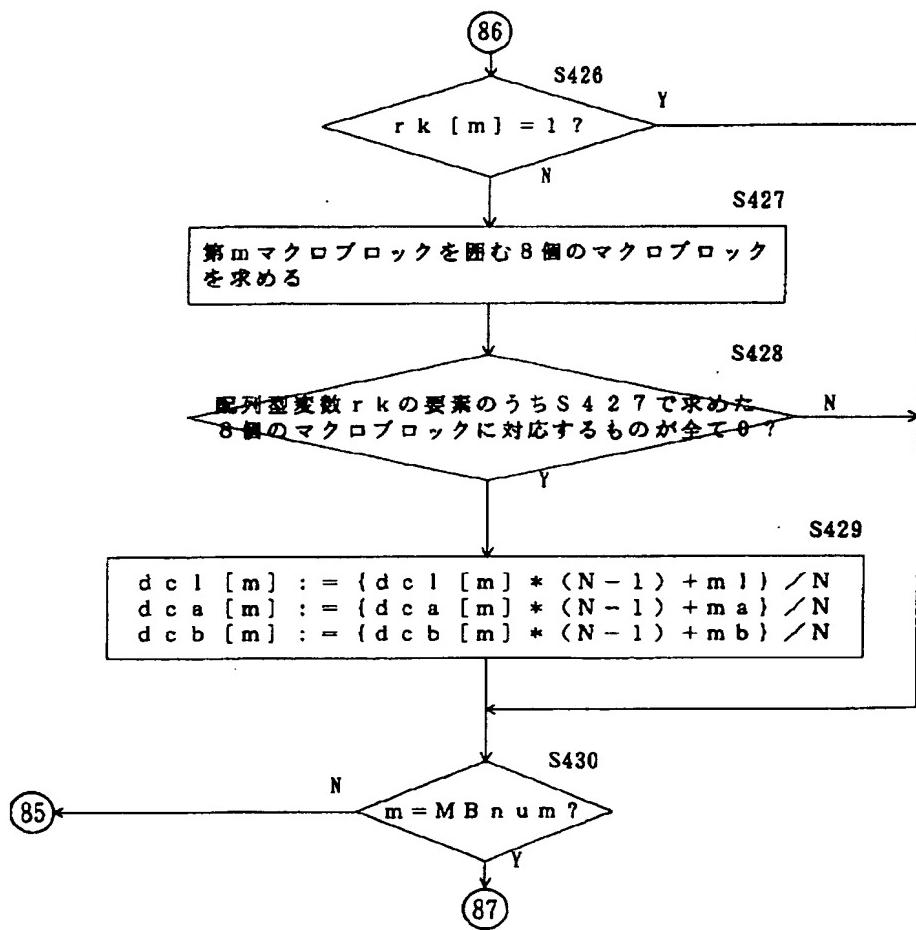
【図12】



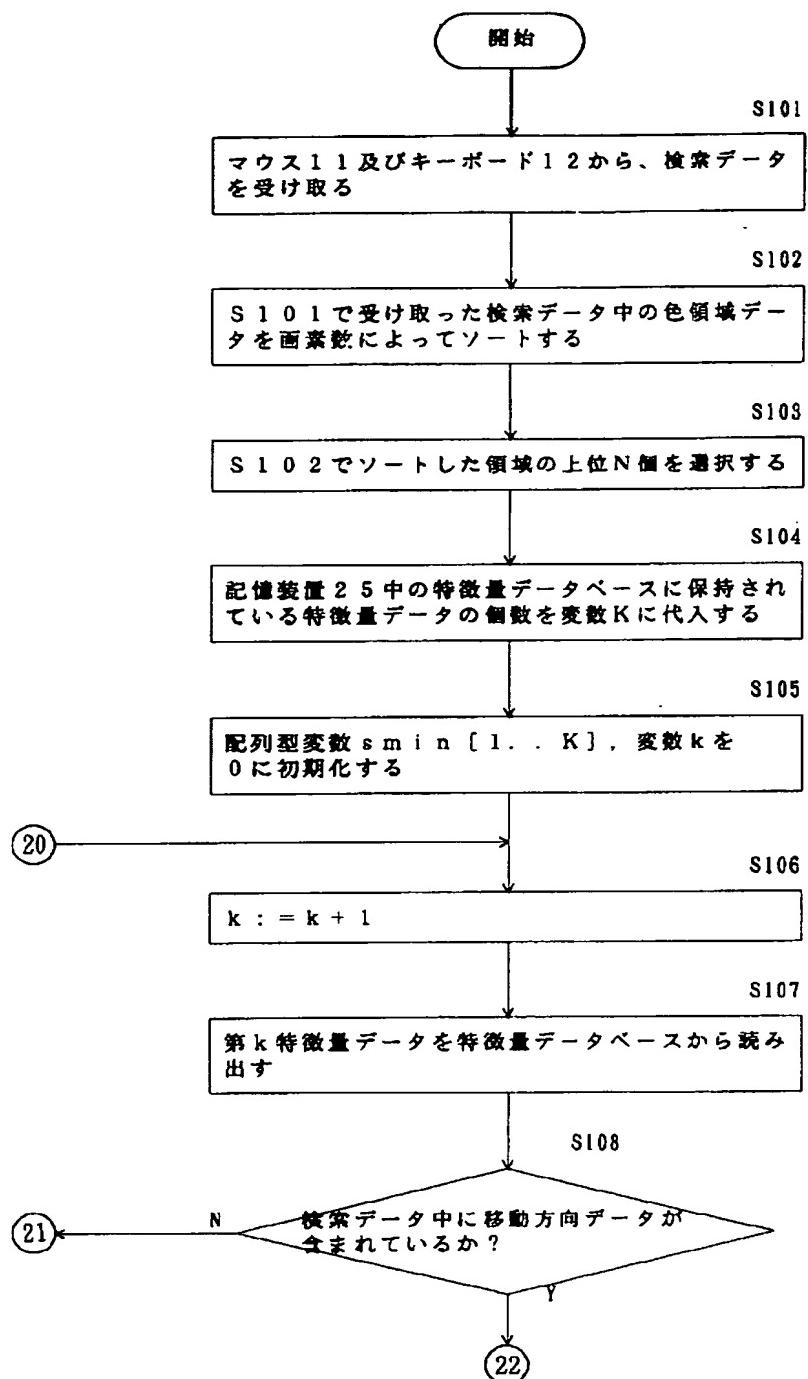
【図13】



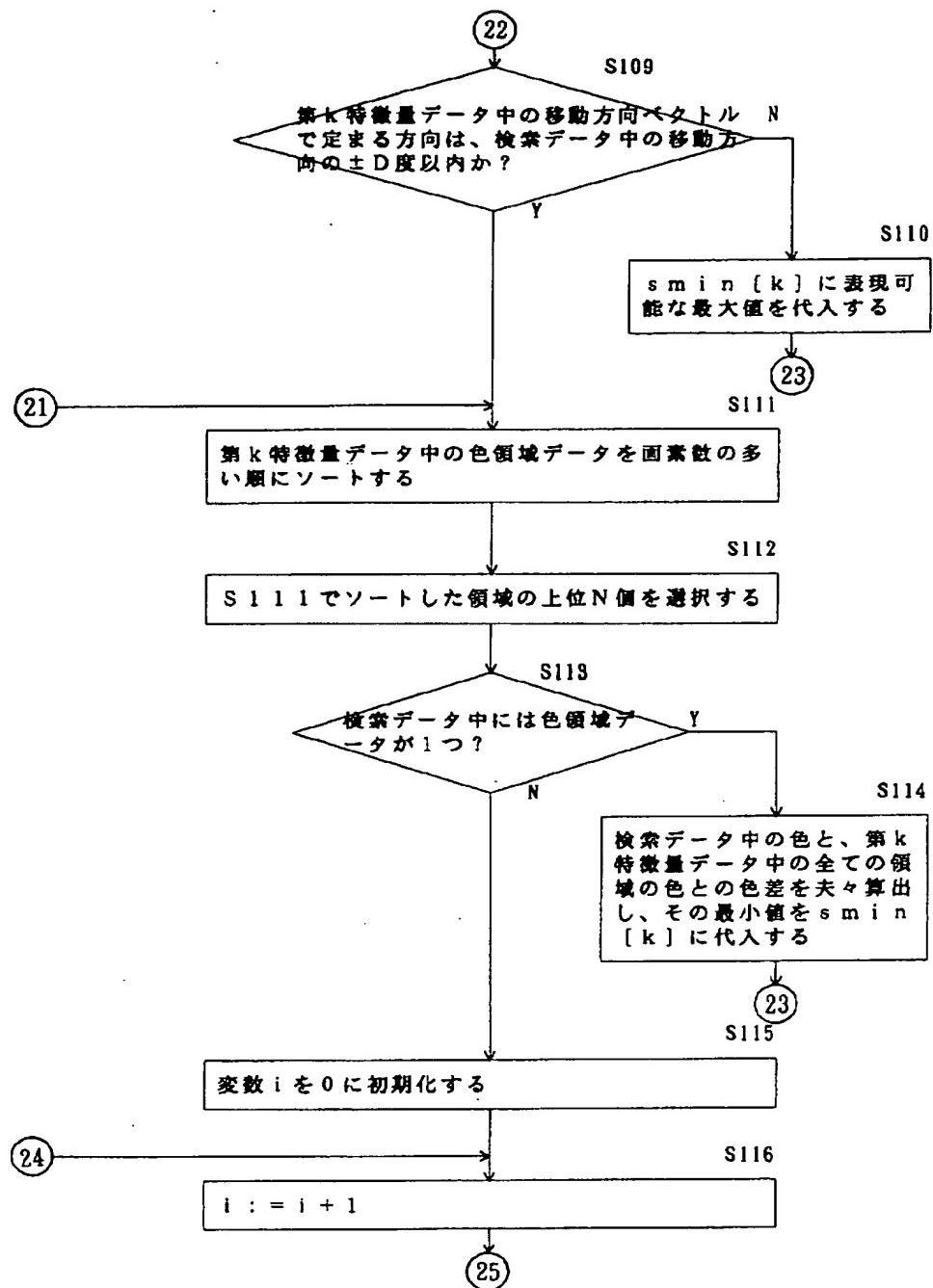
【図31】



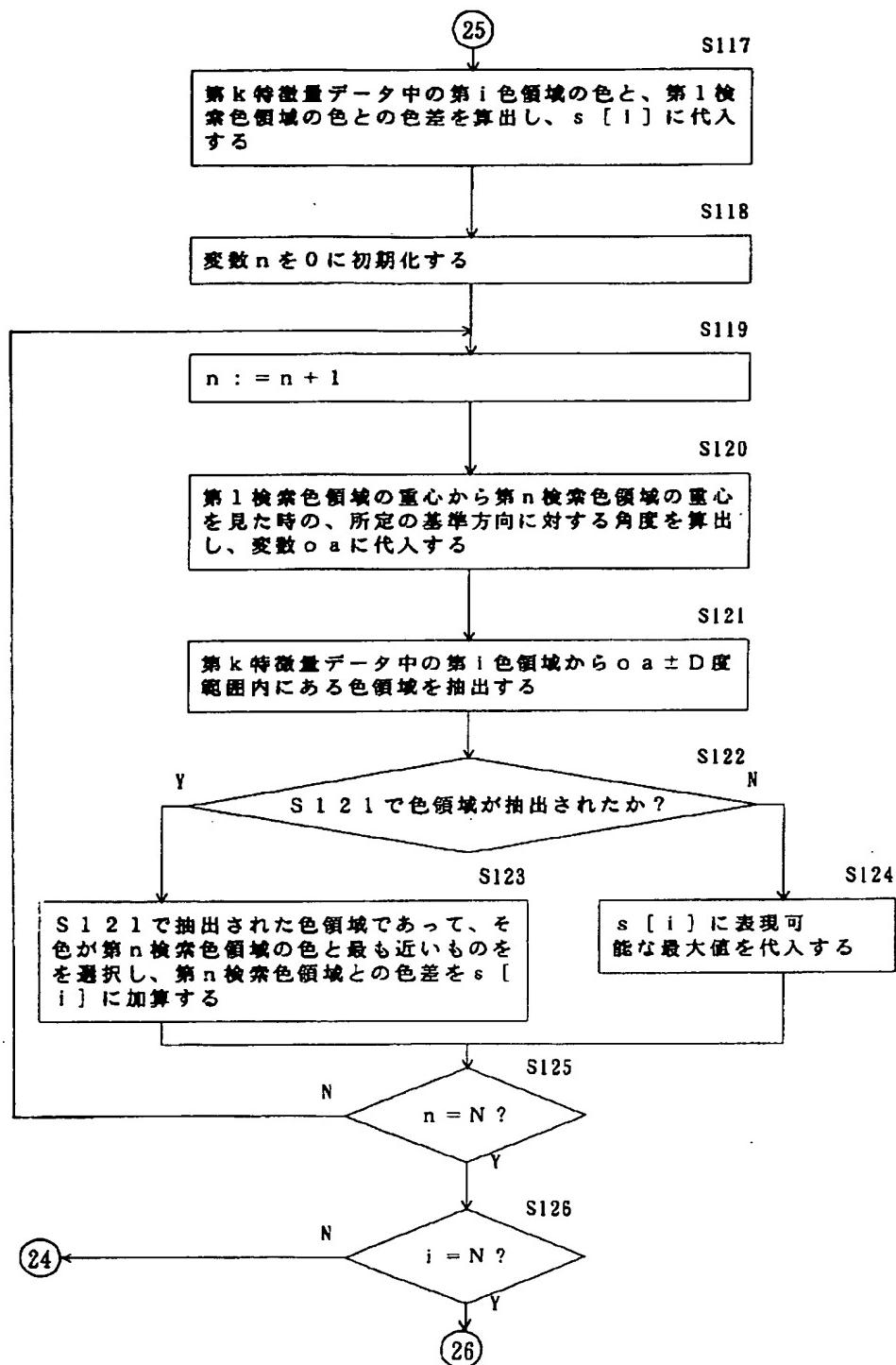
【図14】



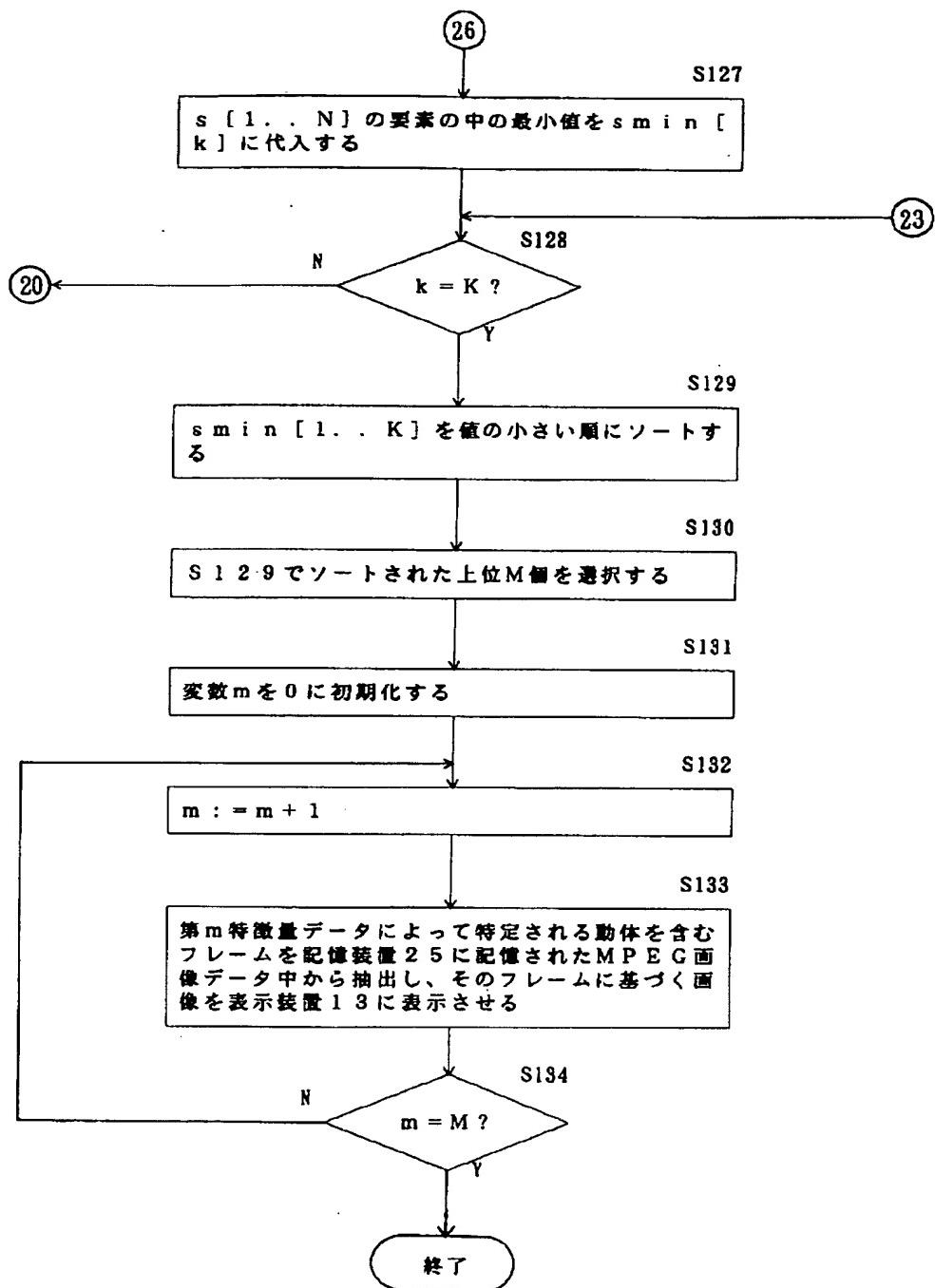
【図15】



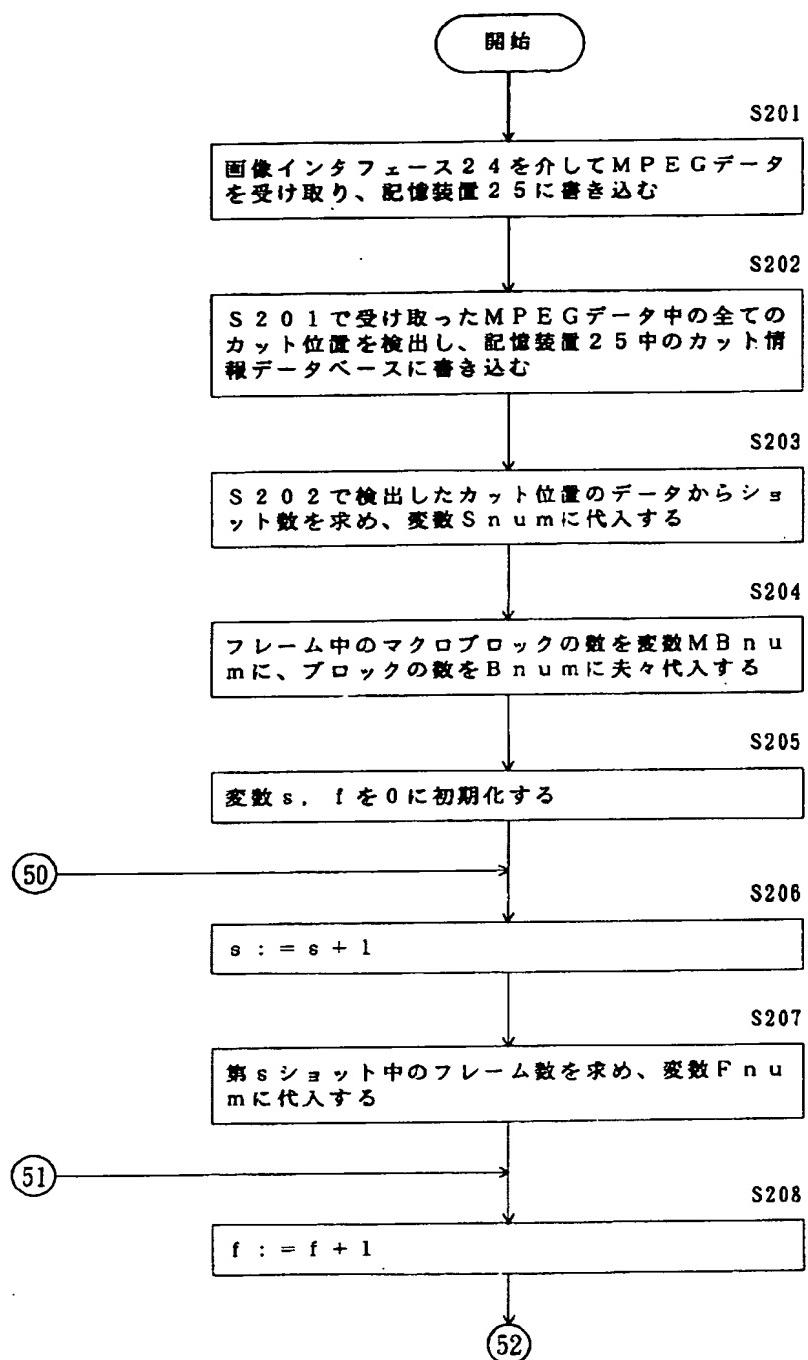
【図16】



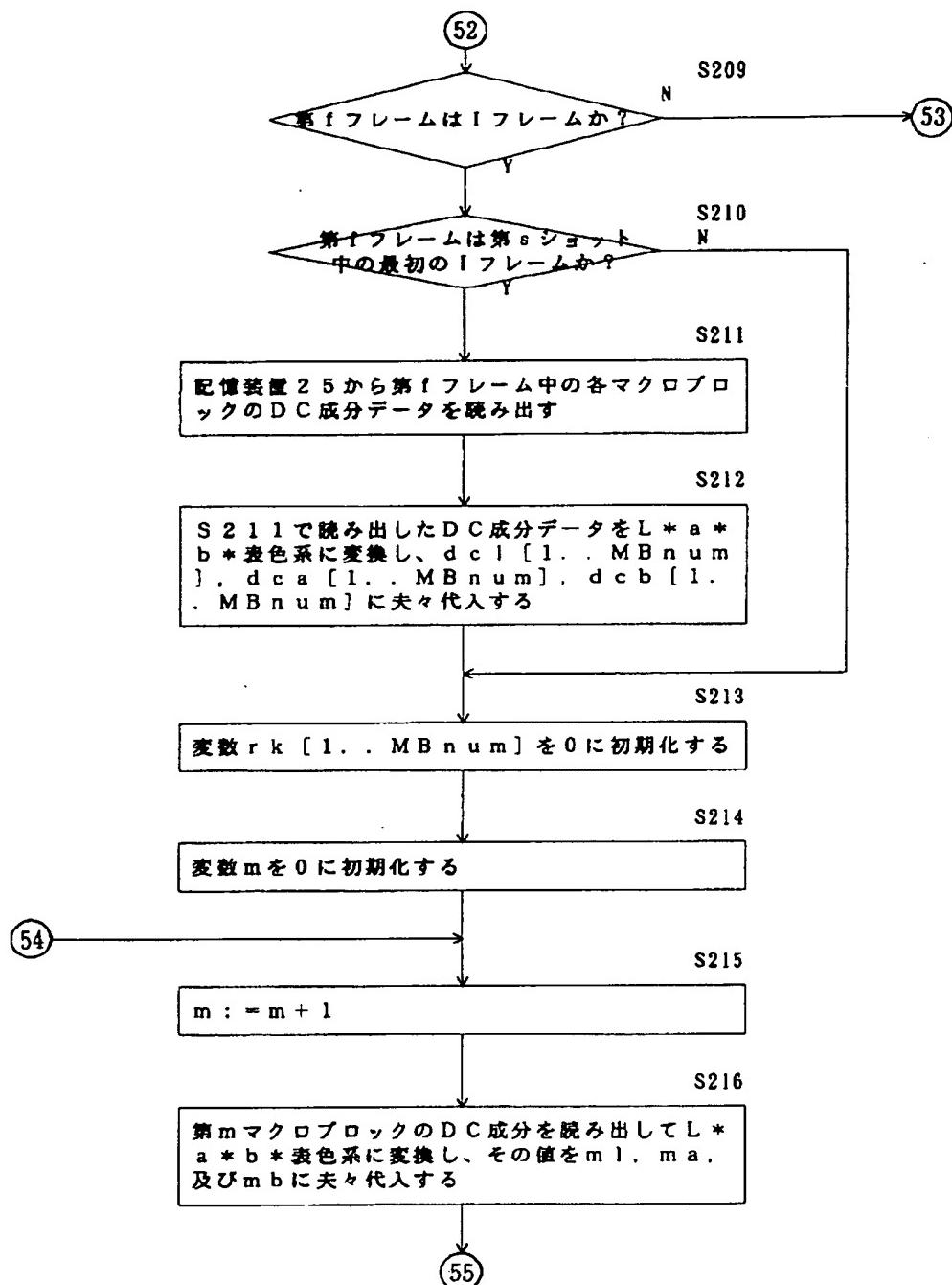
【図17】



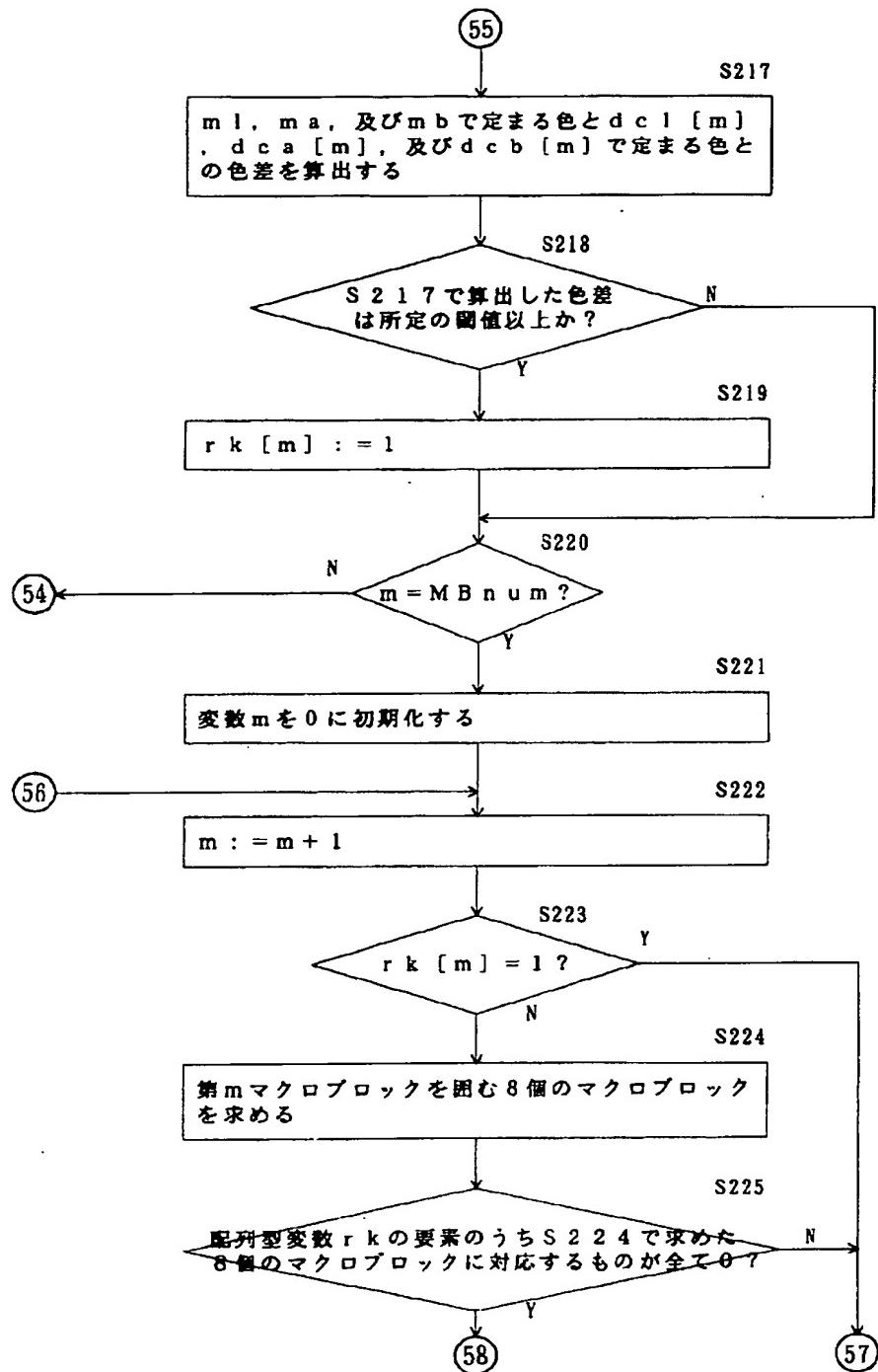
【図18】



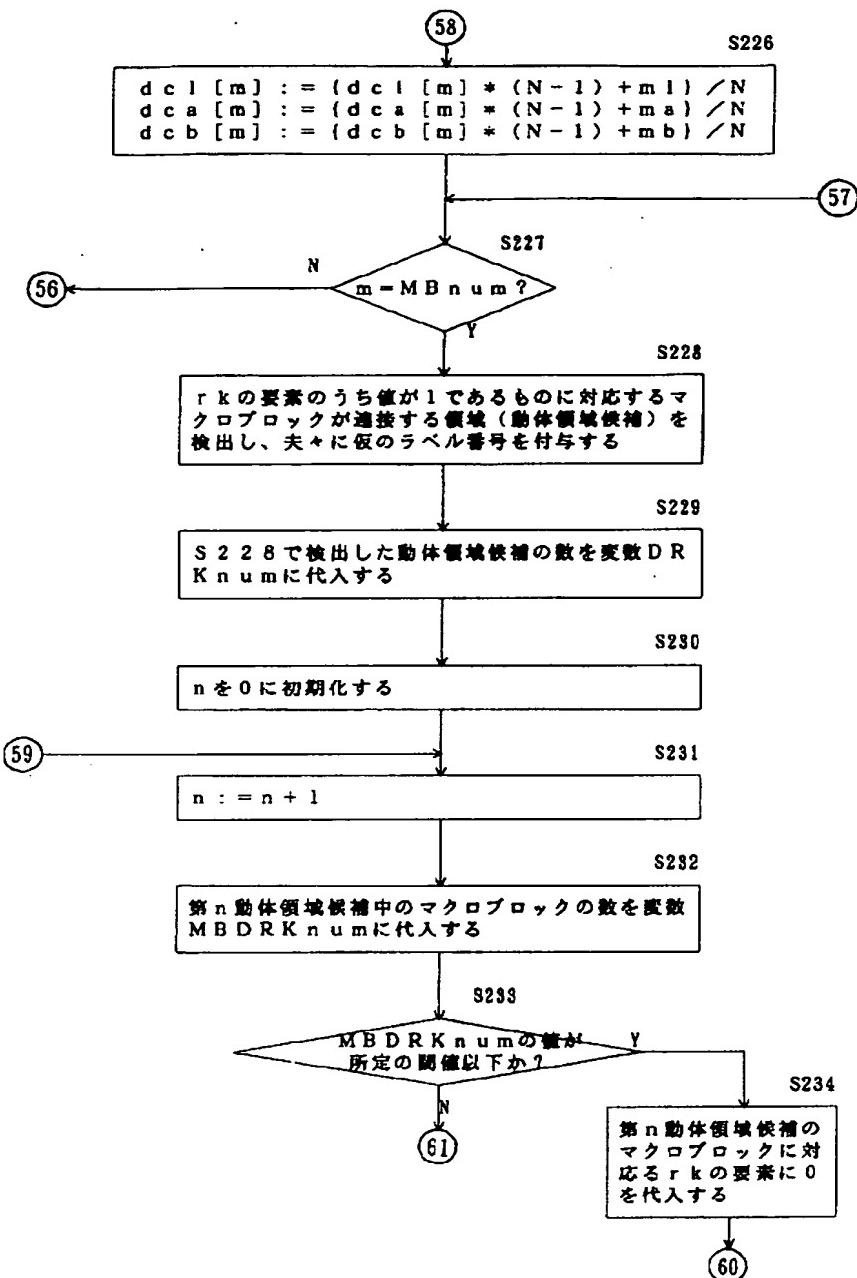
【図19】



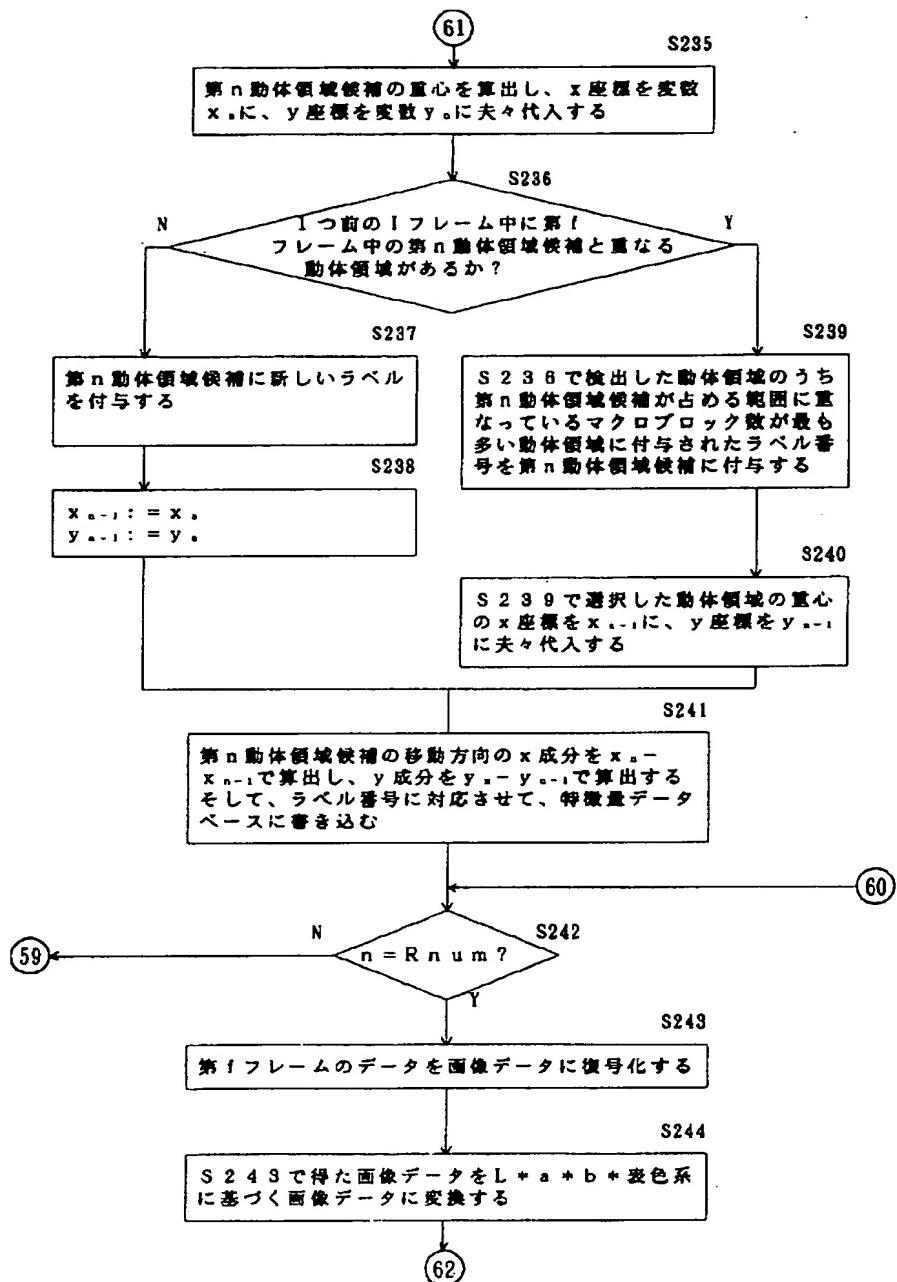
【図20】



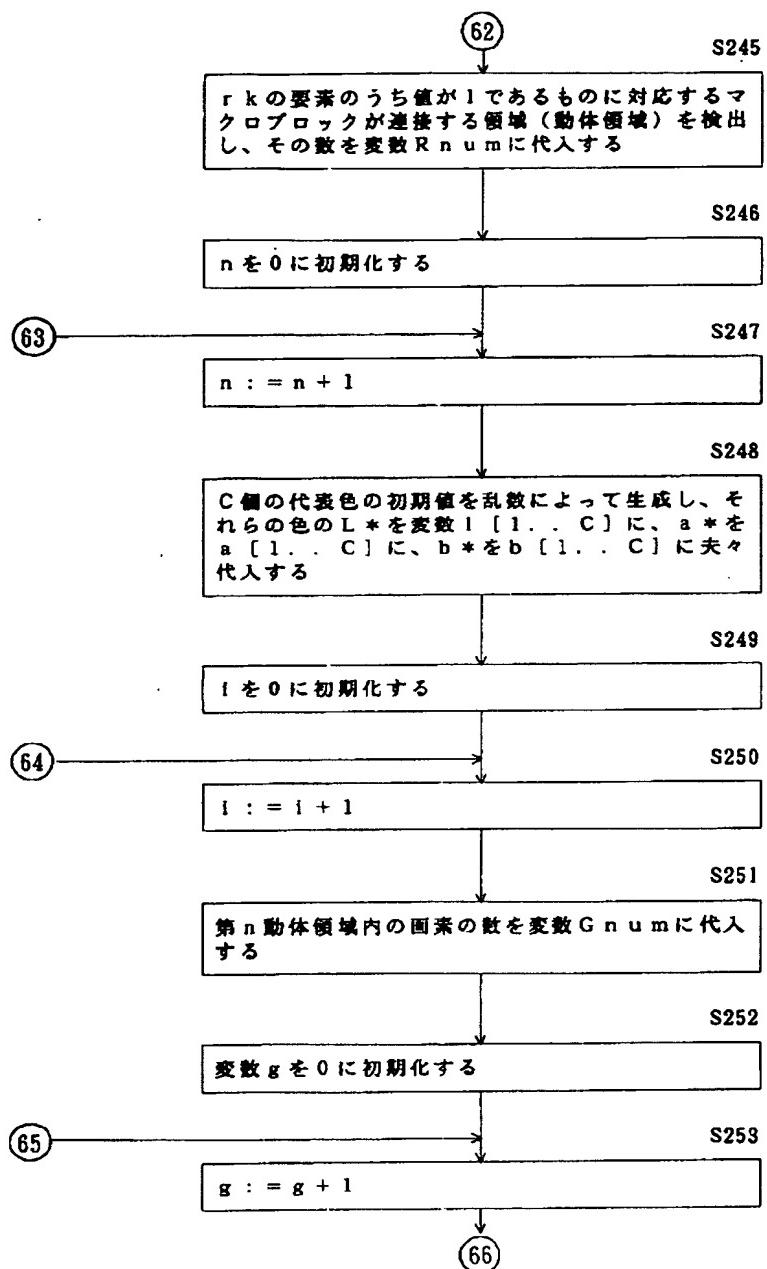
〔图21〕



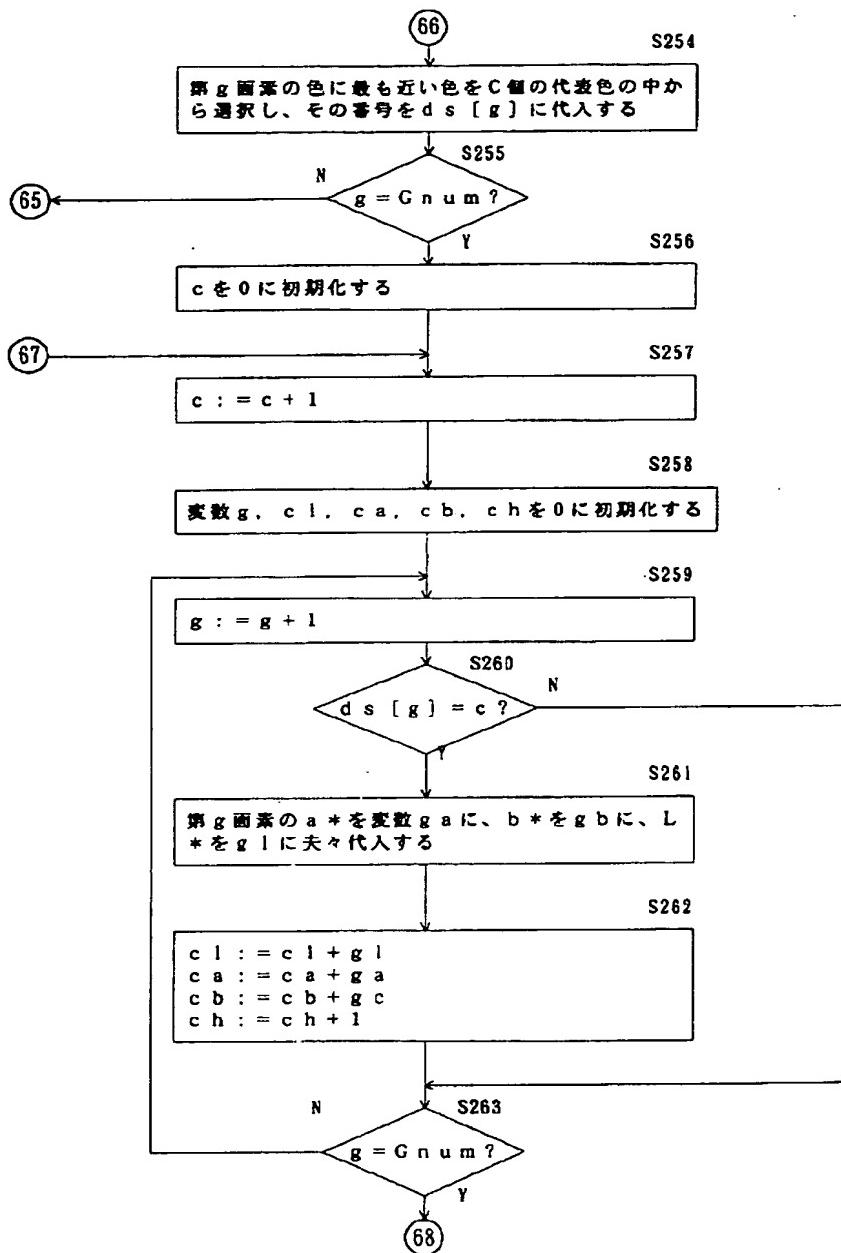
【図22】



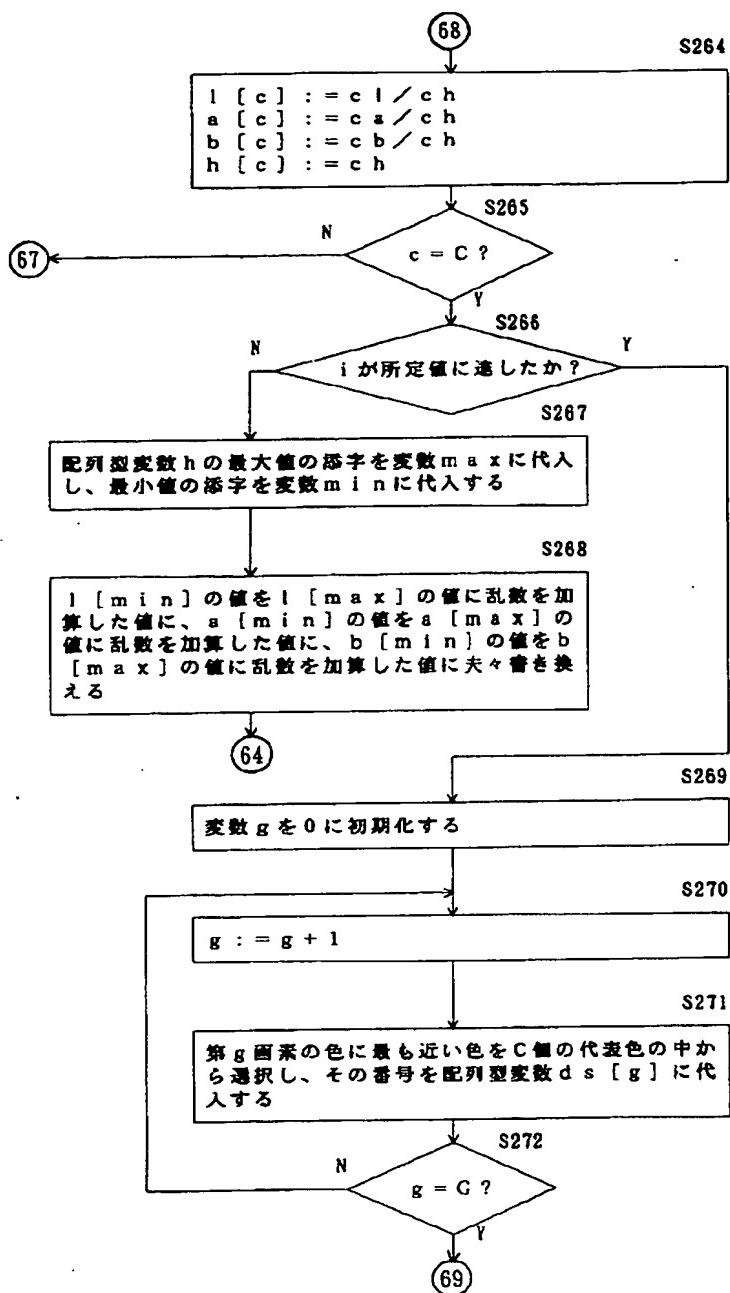
【図23】



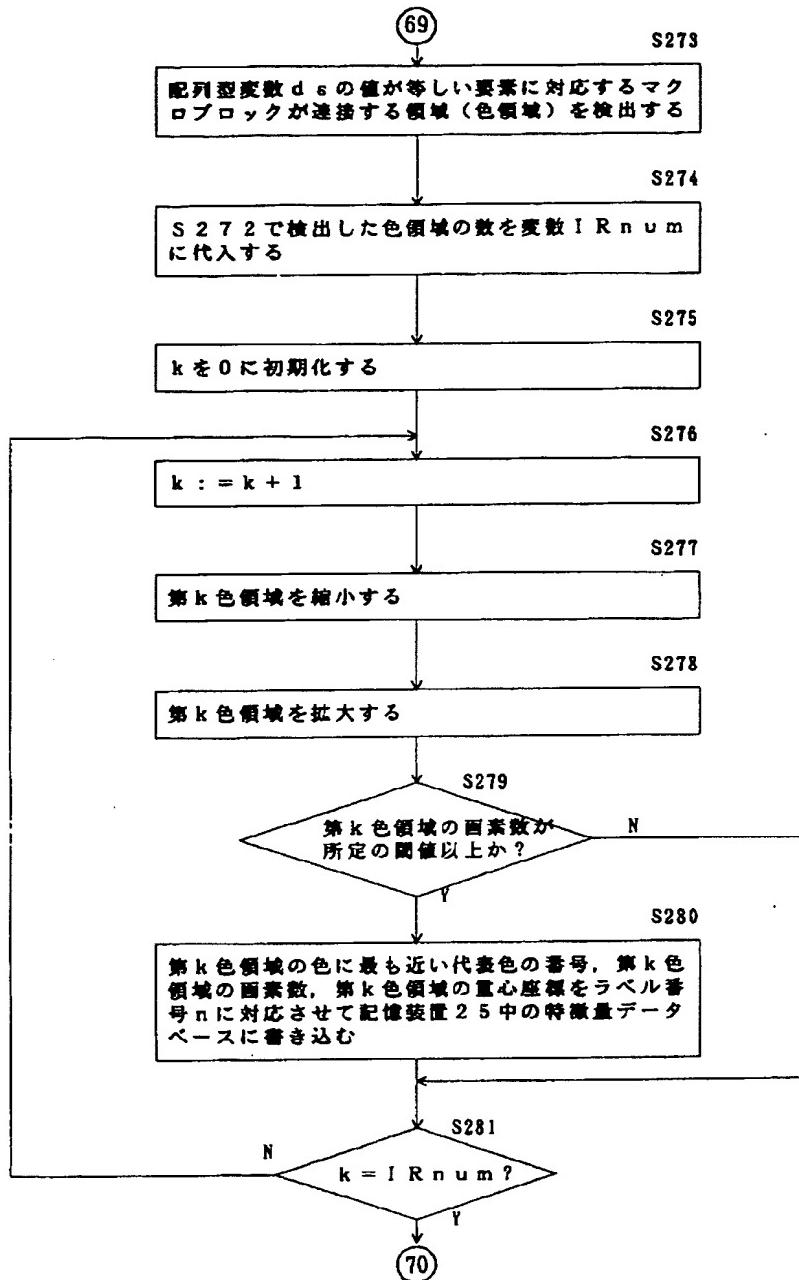
【図24】



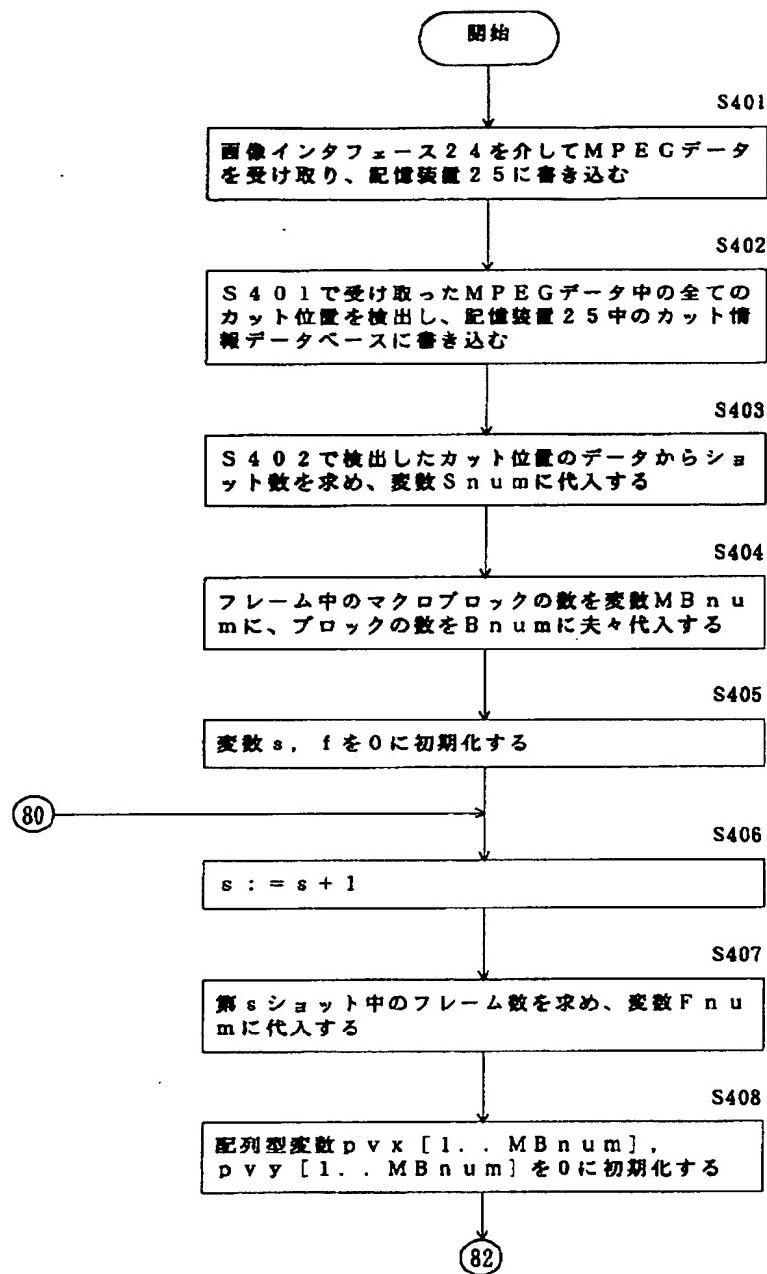
【図25】



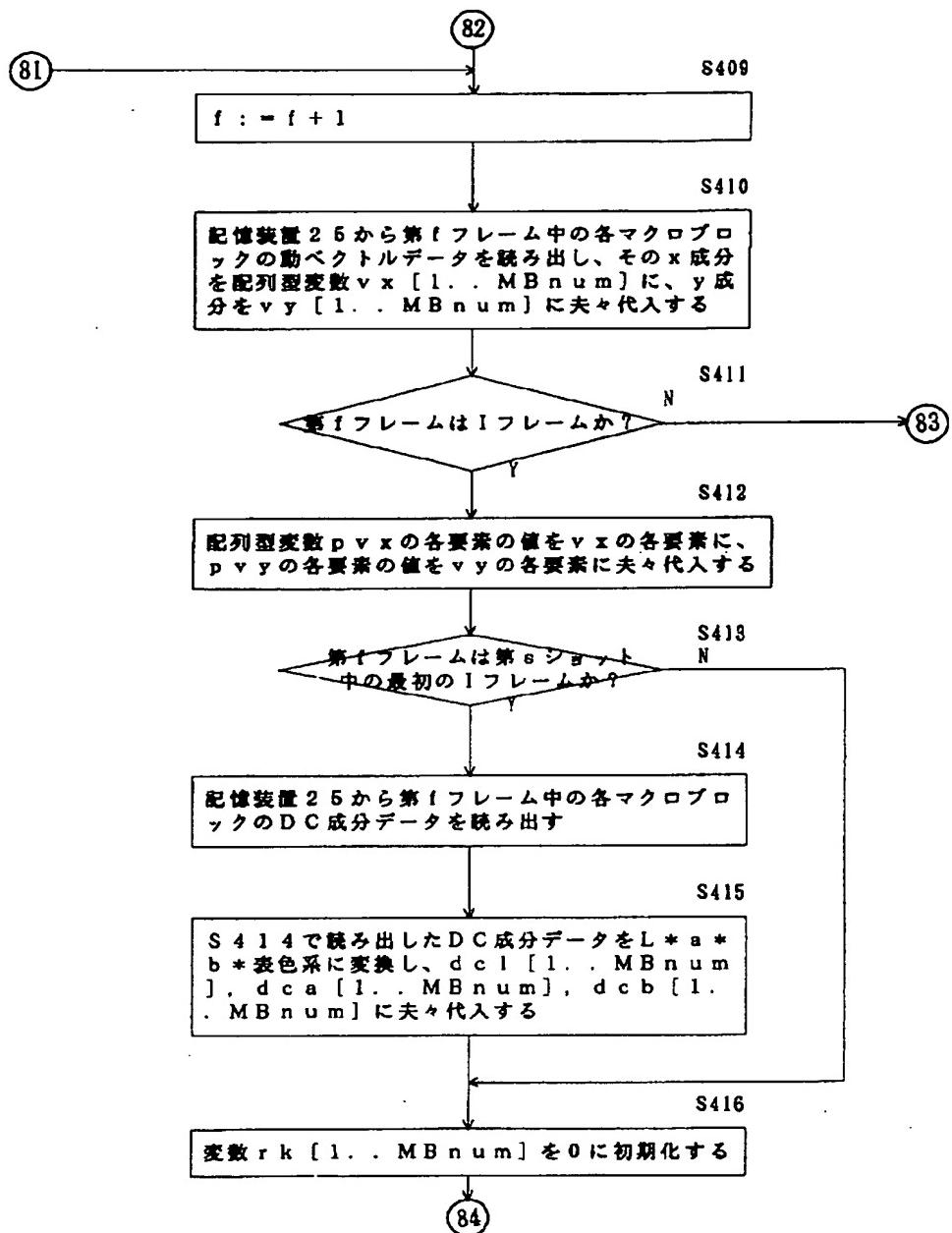
【図26】



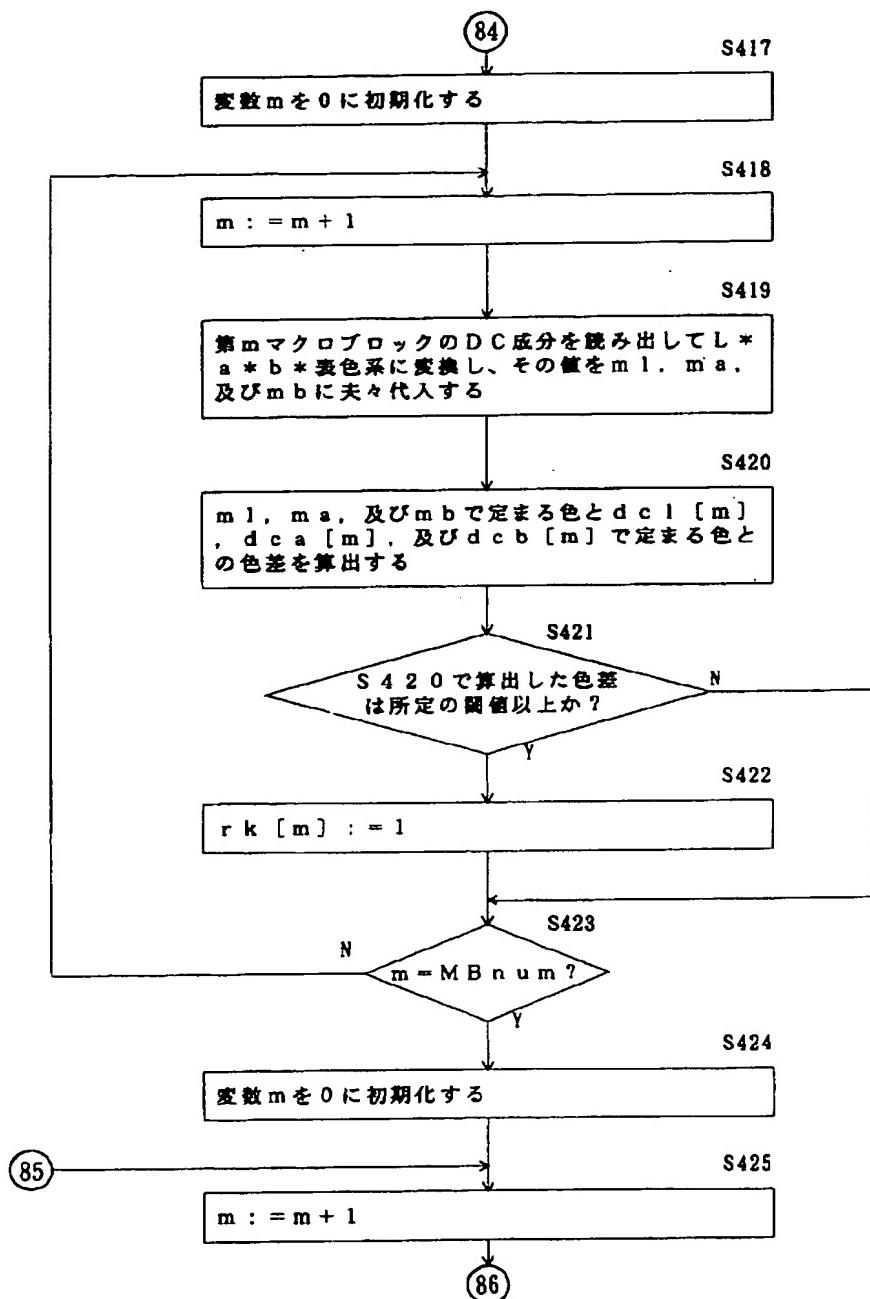
【図28】



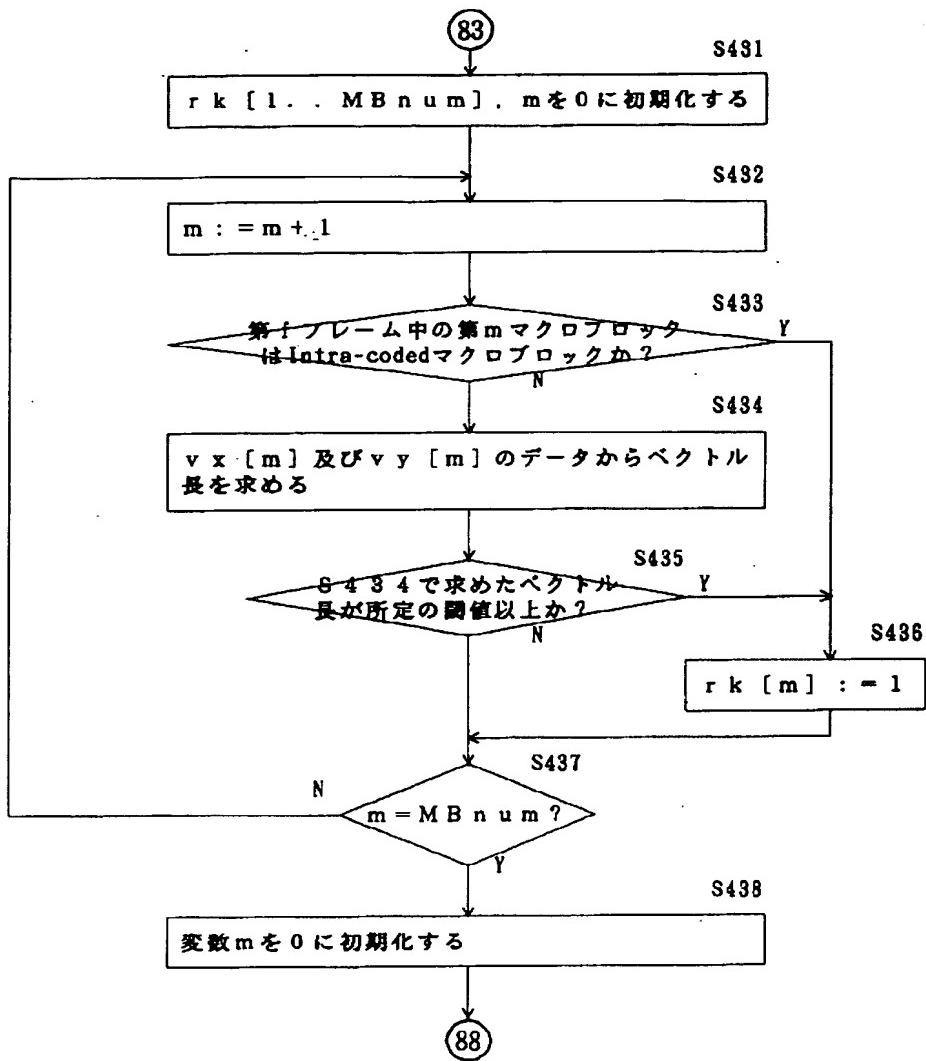
【図29】



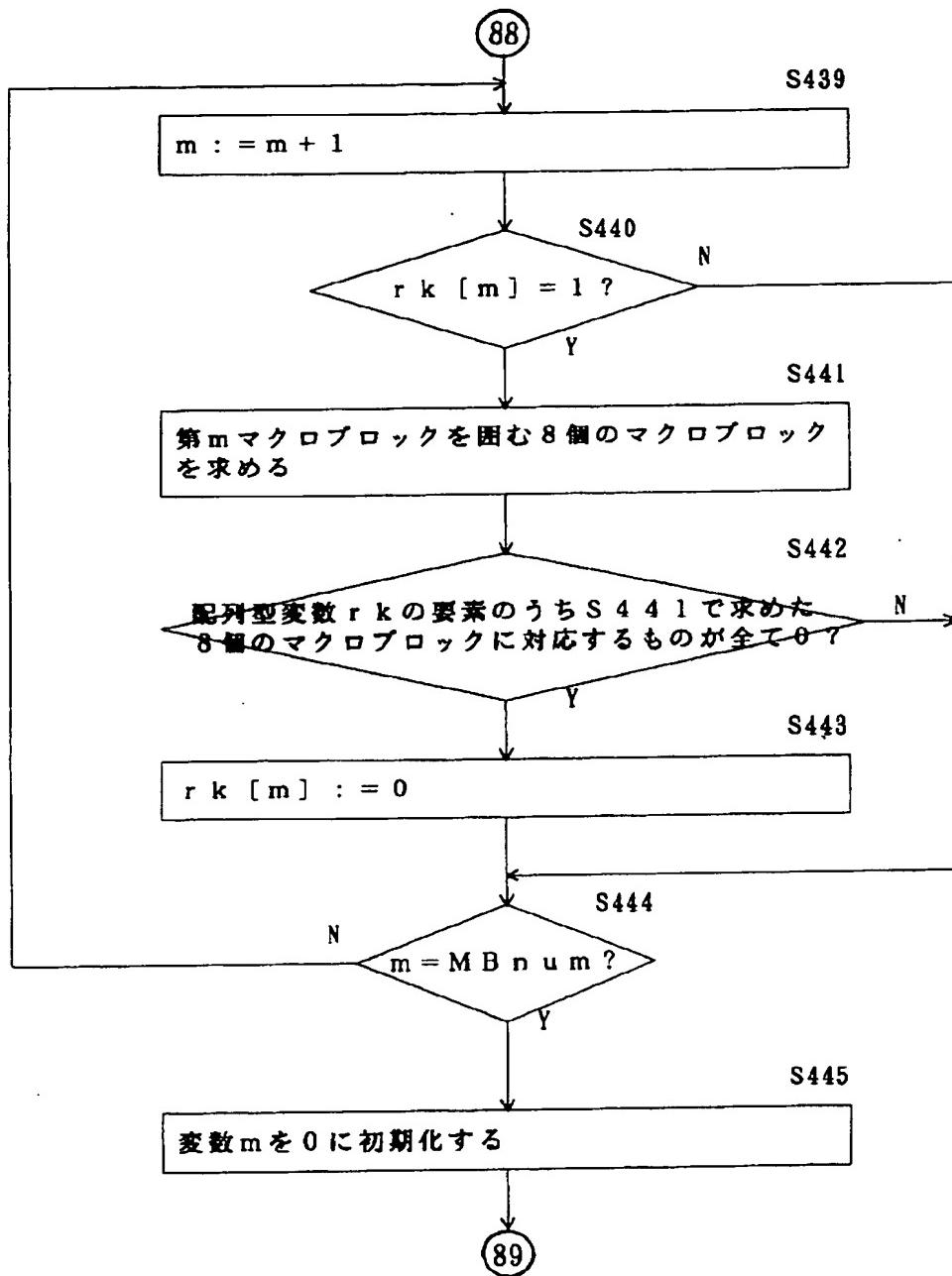
【図30】



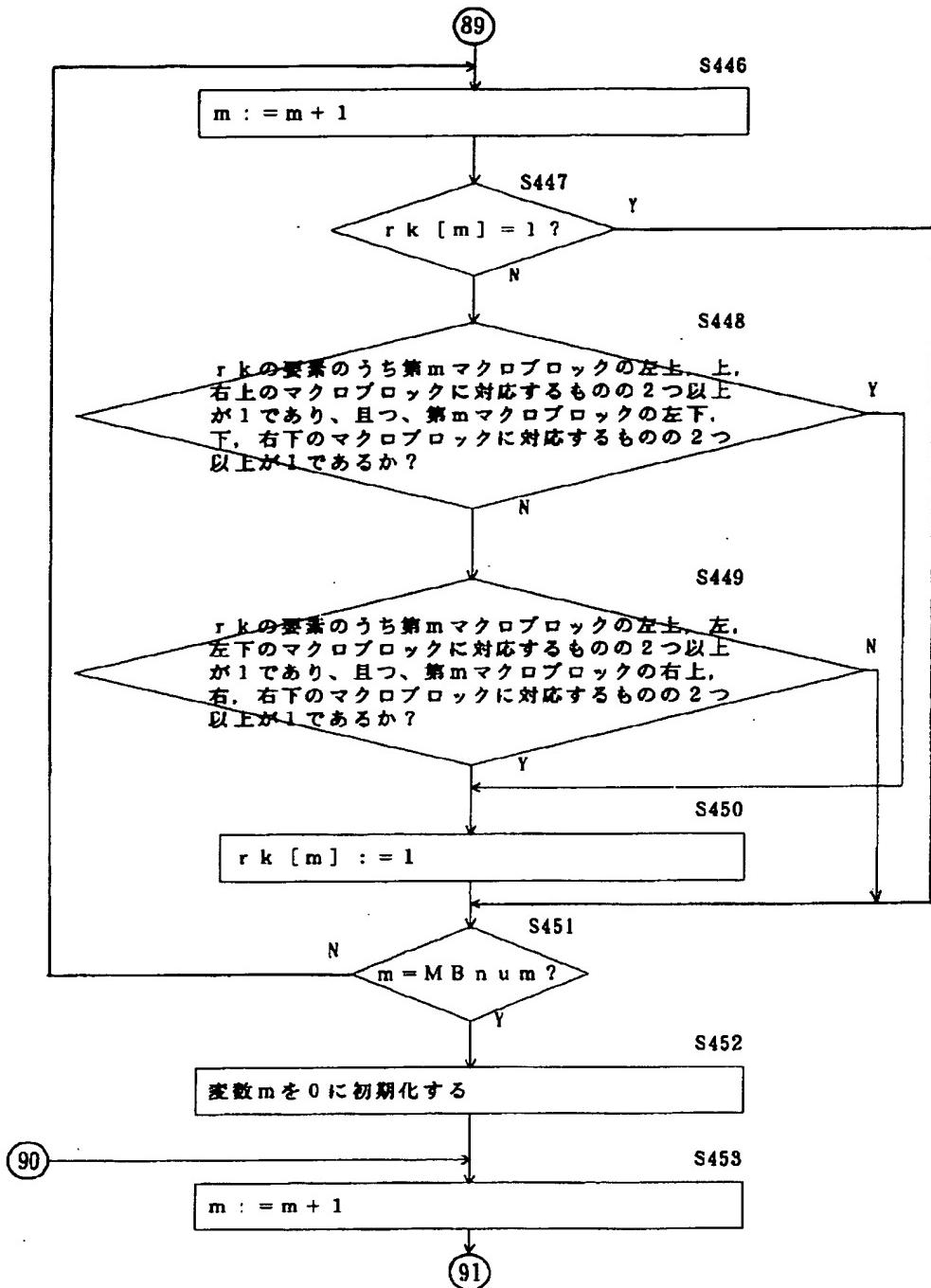
【図32】



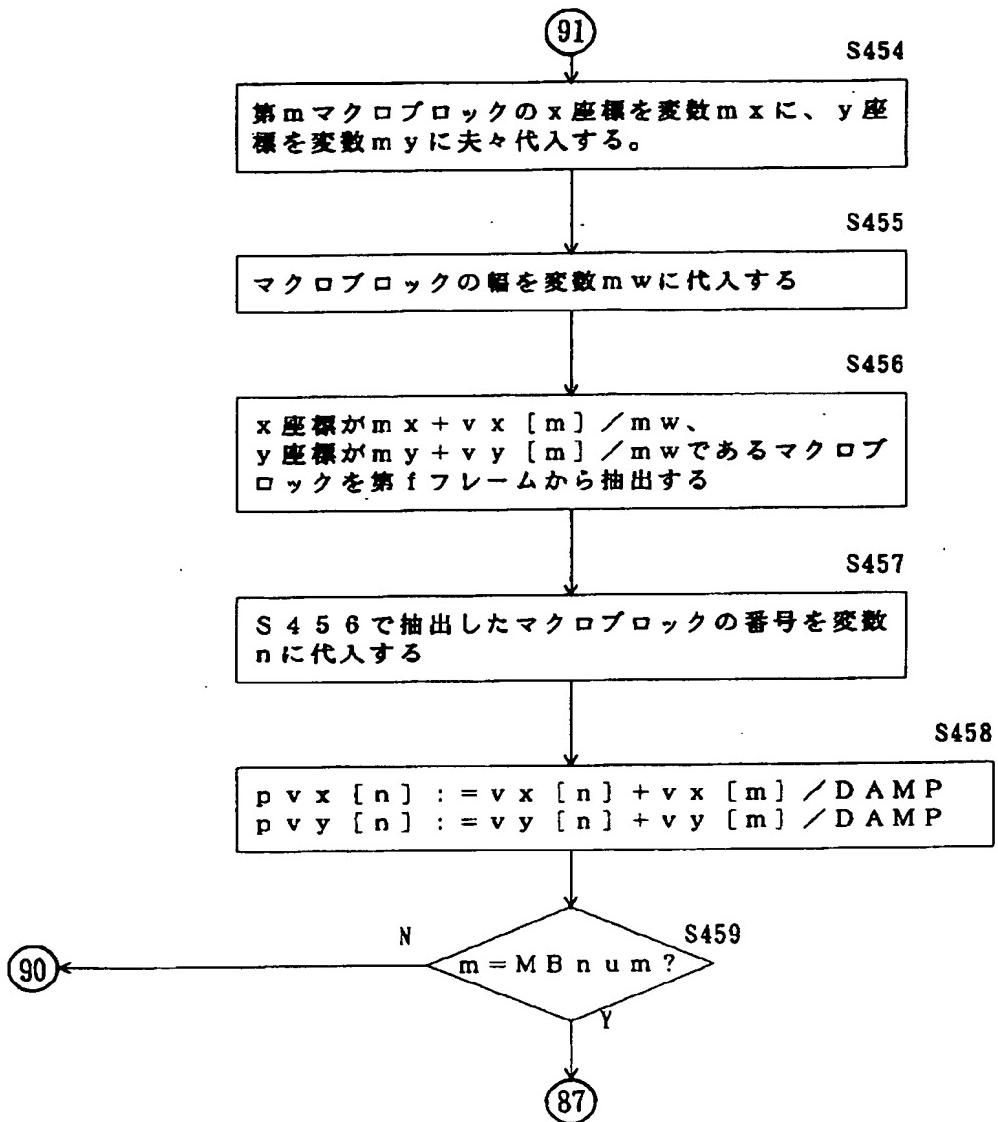
【図33】



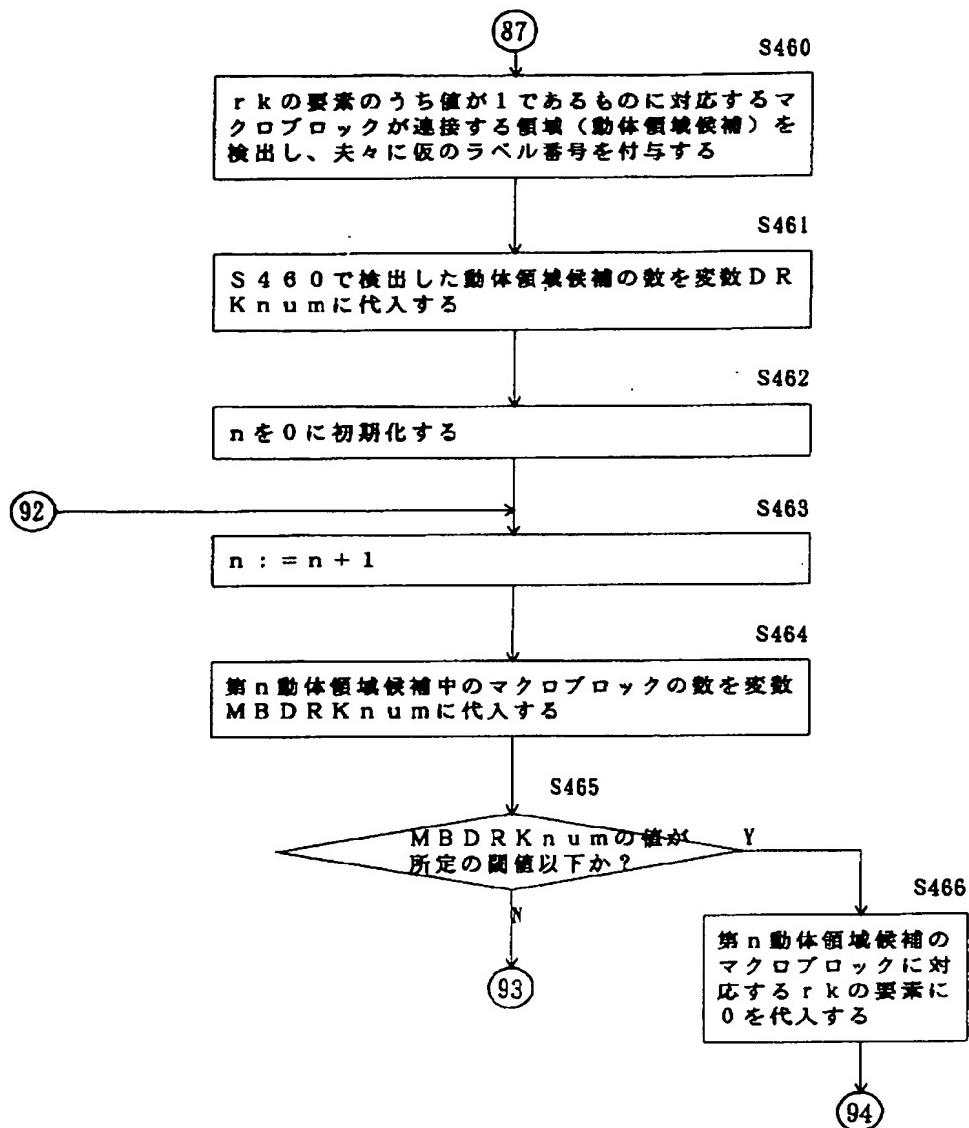
【図34】



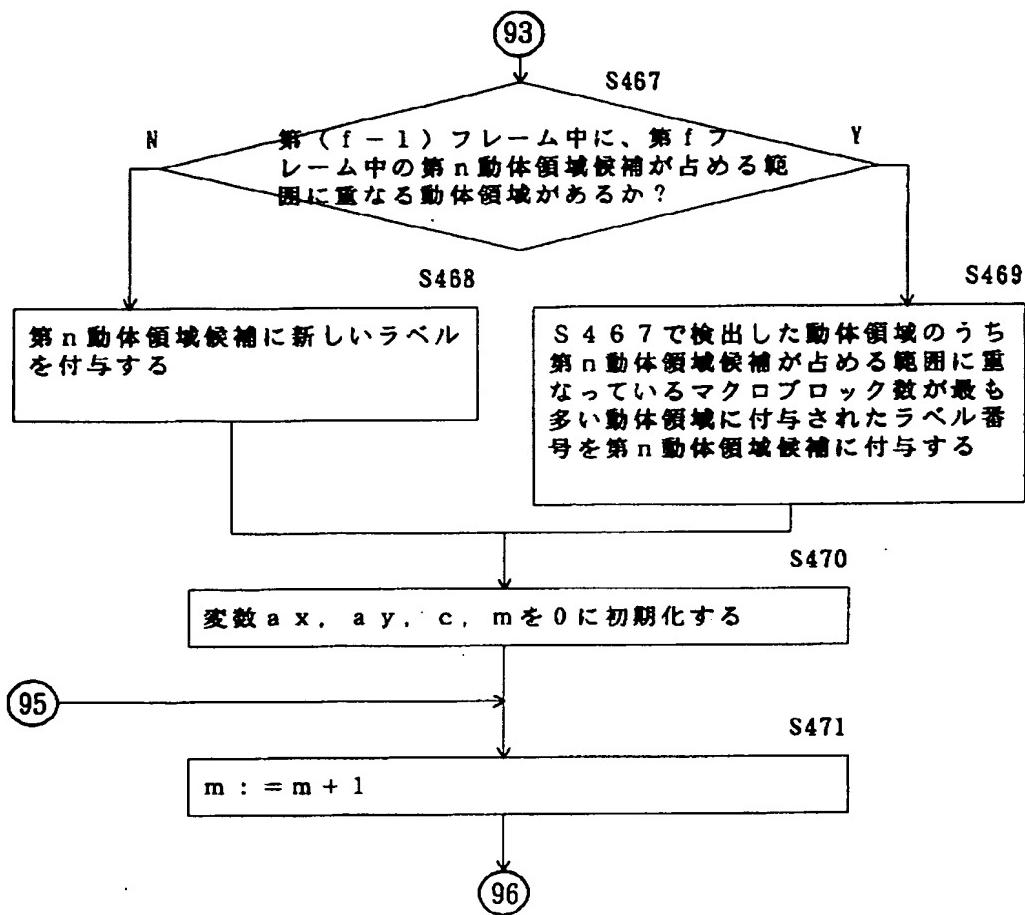
【図35】



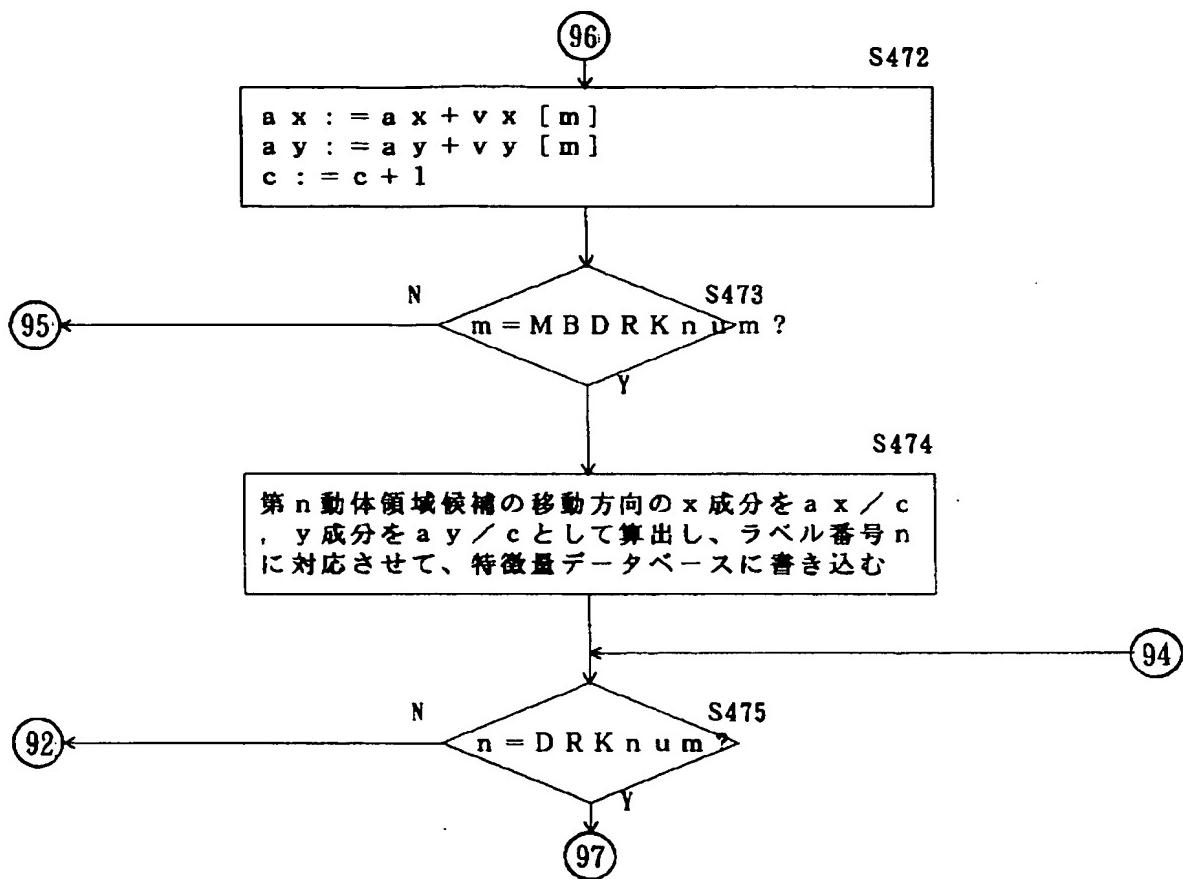
【図36】



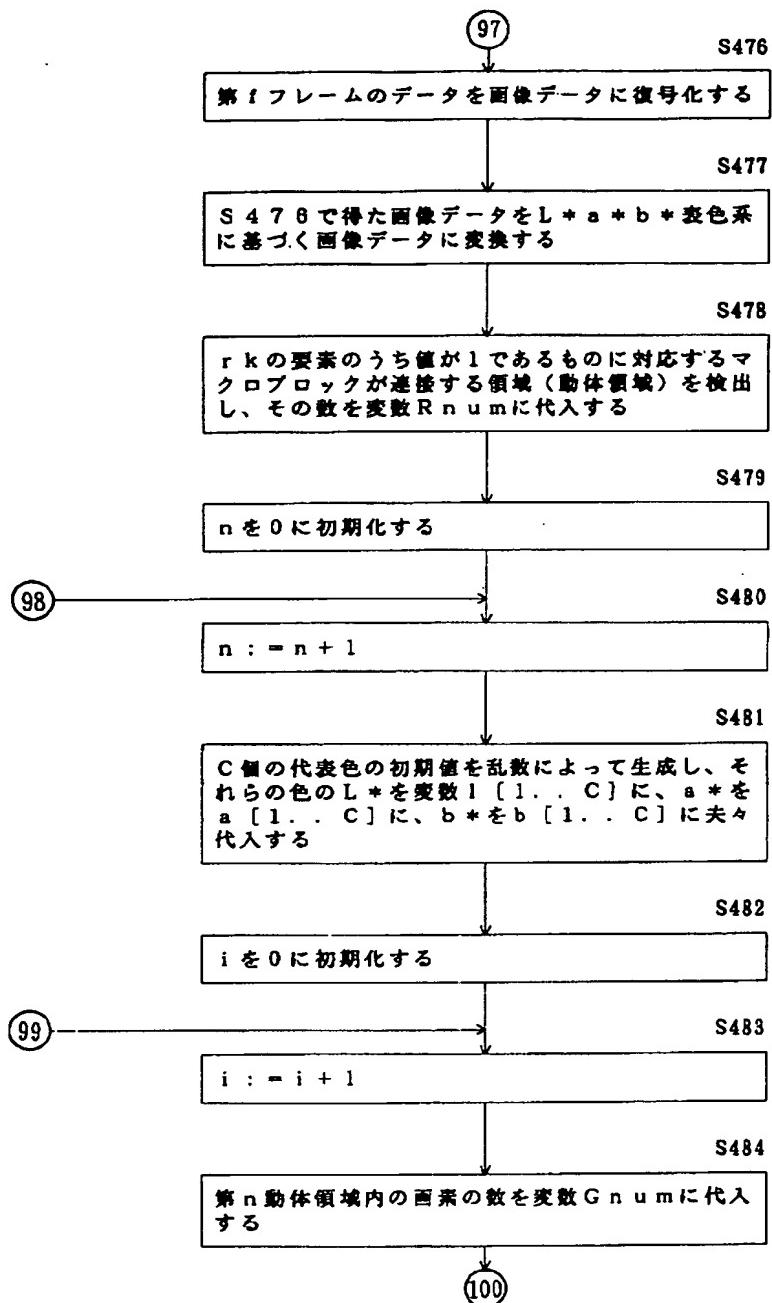
【図37】



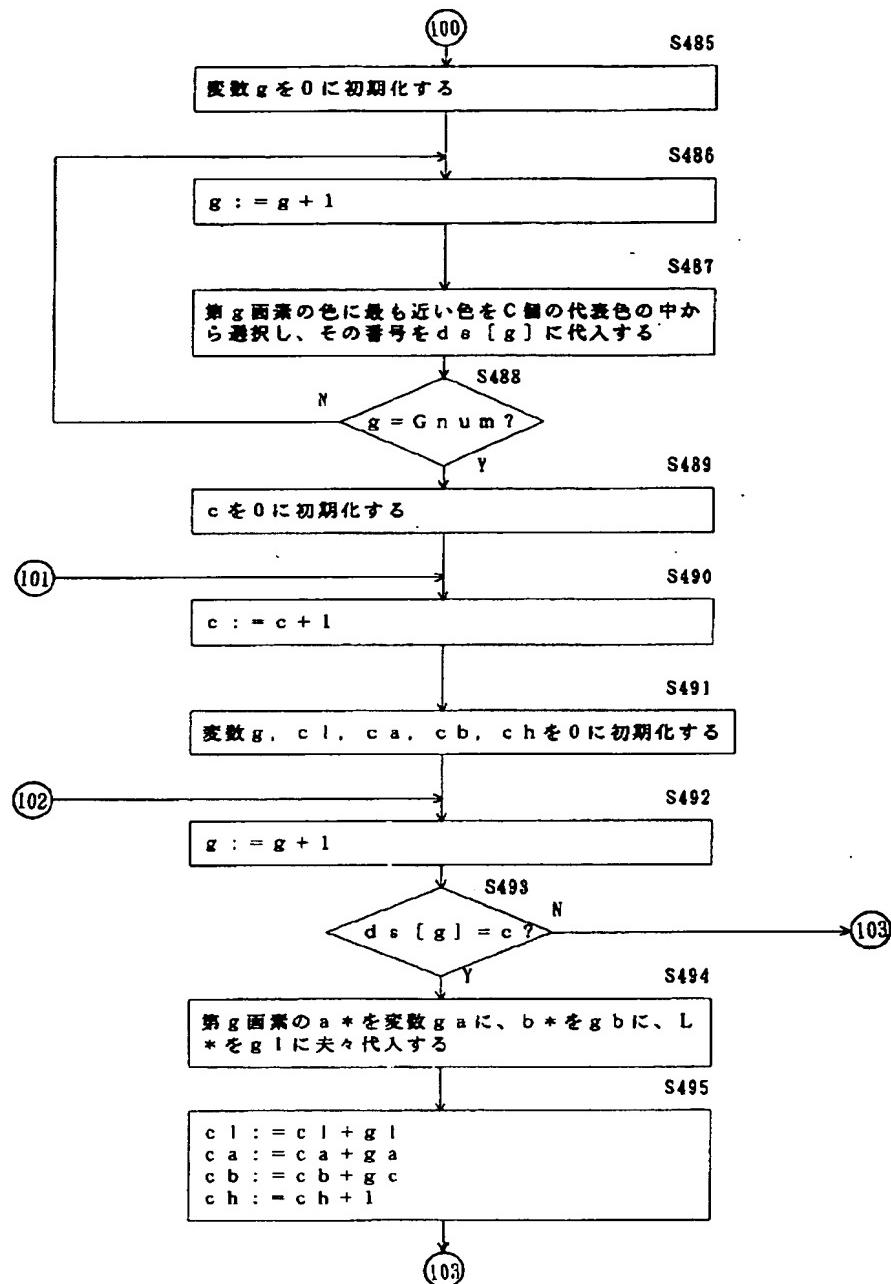
【図38】



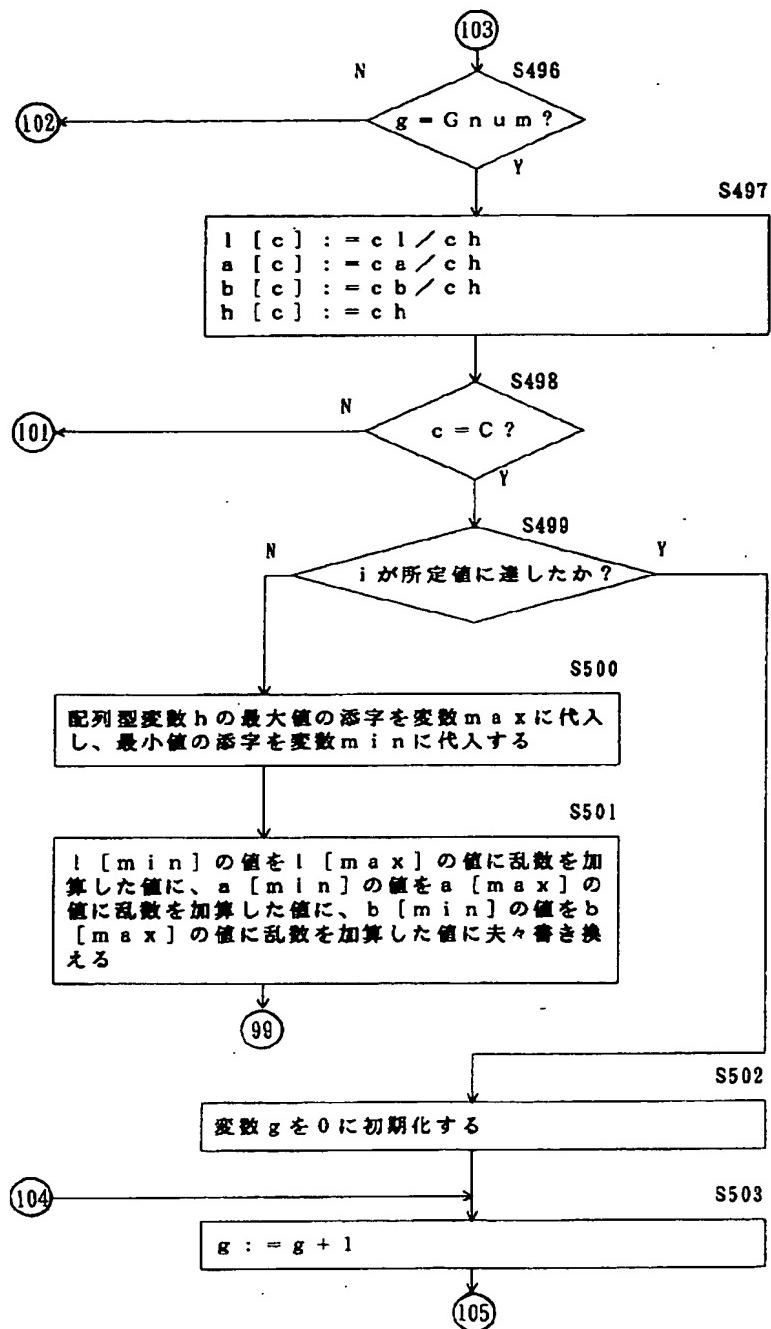
【図39】



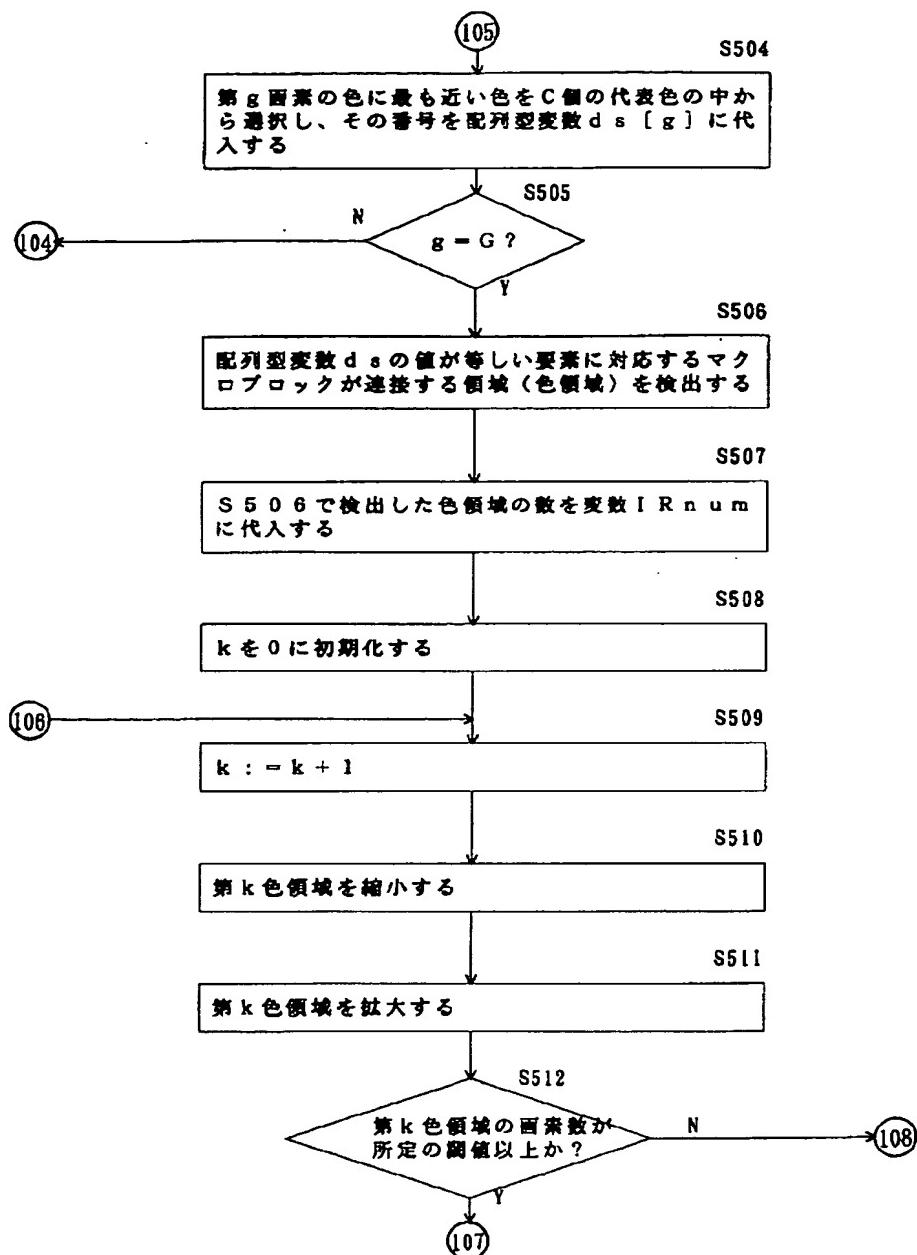
【図40】



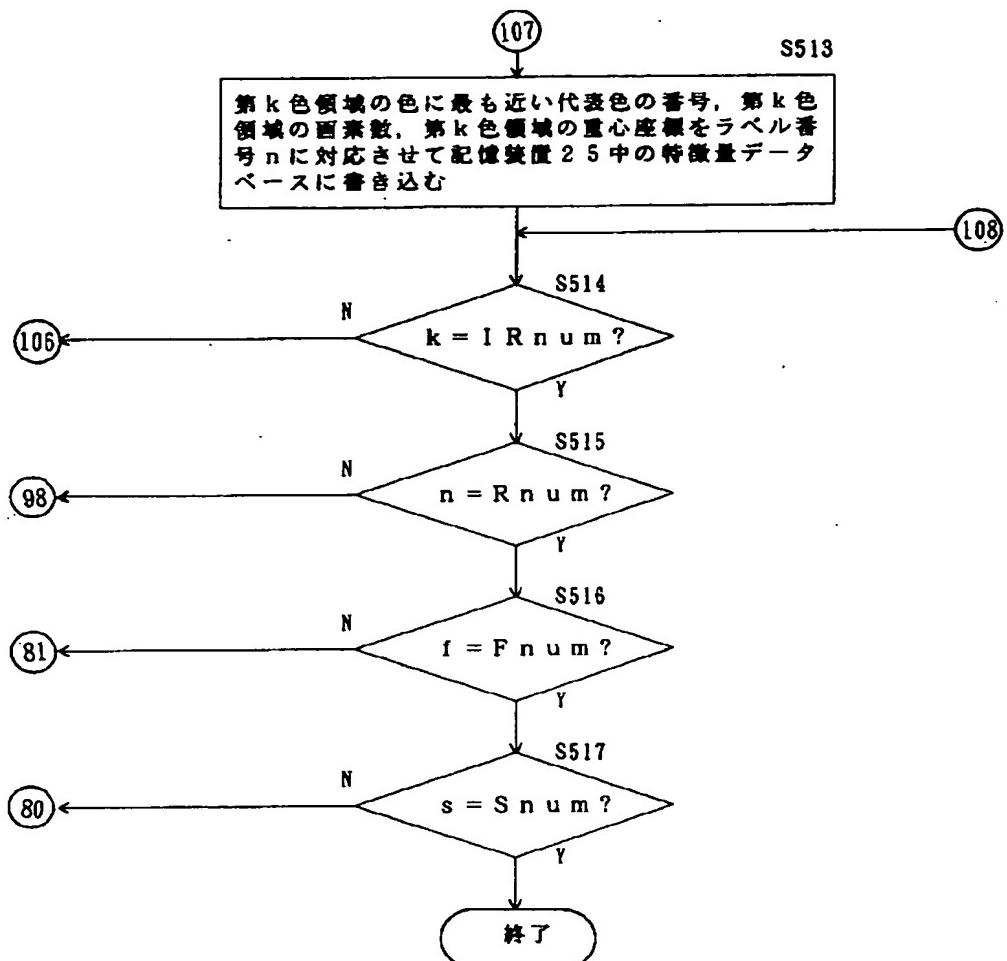
【図4-1】



【図42】



【図43】



**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- BLACK BORDERS**
- IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- FADED TEXT OR DRAWING**
- BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- SKEWED/SLANTED IMAGES**
- COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- GRAY SCALE DOCUMENTS**
- LINES OR MARKS ON ORIGINAL DOCUMENT**
- REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.